

HARDWARE AND SOFTWARE IMPROVEMENTS TO A
PACED DATA ACQUISITION SYSTEM
FOR TURBOMACHINES

Patrick Anthony McCarville

NAVAL POSTGRADUATE SCHOOL

Monterey, California



THESIS

HARDWARE AND SOFTWARE IMPROVEMENTS
TO A PACED DATA ACQUISITION SYSTEM
FOR TURBOMACHINES

by

Patrick Anthony McCarville

June 1981

Thesis Advisor:

R. P. Shreeve

Approved for public release; distribution unlimited

T200050

REPORT DOCUMENTATION PAGE		READ INSTRUCTIONS BEFORE COMPLETING FORM
1. REPORT NUMBER	2. GOVT ACCESSION NO.	3. RECIPIENT'S CATALOG NUMBER
4. TITLE (and Subtitle) Hardware and Software Improvements to a Paced Data Acquisition System for Turbomachines		5. TYPE OF REPORT & PERIOD COVERED Master's Thesis; June 1981
		6. PERFORMING ORG. REPORT NUMBER
7. AUTHOR(s) Patrick Anthony McCarville		8. CONTRACT OR GRANT NUMBER(s)
9. PERFORMING ORGANIZATION NAME AND ADDRESS Naval Postgraduate School Monterey, California 93940		10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS
11. CONTROLLING OFFICE NAME AND ADDRESS Naval Postgraduate School Monterey, California 93940		12. REPORT DATE June 1981
		13. NUMBER OF PAGES 69
14. MONITORING AGENCY NAME & ADDRESS (if different from Controlling Office) Naval Postgraduate School Monterey, California 93940		15. SECURITY CLASS. (of this report) Unclassified
		15a. DECLASSIFICATION/DOWNGRADING SCHEDULE
16. DISTRIBUTION STATEMENT (of this Report) Approved for public release; distribution unlimited		
17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)		
18. SUPPLEMENTARY NOTES		
19. KEY WORDS (Continue on reverse side if necessary and identify by block number) Programmable; synchronized sampling; digital phase lock loop, direct memory access		
20. ABSTRACT (Continue on reverse side if necessary and identify by block number) Modification of the phase lock loop synchronizing circuits and of the method of input/output communication used in a syn- chronized data sampling system, are reported. A device known as PACER which used an analog phase lock loop for synchronization and produced a non linear set of synchronizing pulses, was modified to use a CMOS digital phase lock loop, resulting in a linear set of pulses. The associated programming which controlled the data		

acquisition process and sequencing, was changed to use the direct memory access feature of the system computer. This enabled data, from high response pressure transducers mounted in a turbomachine, to be taken once every rotor revolution rather than once every ten revolutions. A user's manual for paced data acquisition is included.

Approved for public release; distribution unlimited

Hardware and Software Improvements
to a Paced Data Acquisition System
for Turbomachines

by

Patrick Anthony McCarville
Lieutenant Commander, United States Navy
B.S., University of New Mexico, 1972

Submitted in partial fulfillment of the
requirements for the degree of

MASTER OF SCIENCE IN AERONAUTICAL ENGINEERING

from the

NAVAL POSTGRADUATE SCHOOL
June 1981

ABSTRACT

Modification of the phase lock loop synchronizing circuits and of the method of input/output communication used in a synchronized data sampling system, are reported. A device known as PACER which used an analog phase lock loop for synchronization and produced a non linear set of synchronizing pulses, was modified to use a CMOS digital phase lock loop, resulting in a linear set of pulses. The associated programming which controlled the data acquisition process and sequencing, was changed to use the direct memory access feature of the system computer. This enabled data, from high response pressure transducers mounted in a turbomachine, to be taken once every rotor revolution rather than once every ten revolutions. A user's manual for paced data acquisition is included.

TABLE OF CONTENTS

I.	INTRODUCTION -----	10
II.	PACED DATA ACQUISITION SYSTEM DESCRIPTION -----	13
A.	GENERAL DESCRIPTION -----	13
B.	SYSTEM SOFTWARE -----	15
1.	RTE-IVB Operating System -----	15
2.	System Test and Operation Program -----	16
C.	SYSTEM HARDWARE -----	17
1.	Hewlett Packard HP21 MX Computer -----	17
2.	Hewlett Packard HP5610 A A/D Converter ---	18
3.	PACER -----	19
III.	CHANGES TO PACER HARDWARE -----	20
IV.	CHANGE TO ACQUISITION SOFTWARE -----	22
A.	METHODS OF INPUT/OUTPUT -----	22
B.	INCORPORATION OF DMA -----	22
V.	RESULTS -----	24
A.	LINEARITY TEST -----	24
B.	AUTO LOCK-ON TEST -----	25
C.	TEST OF ACQUISITION TIME -----	25
VI.	CONCLUSIONS AND RECOMMENDATIONS -----	27
APPENDIX A:	HARDWARE DESIGN DETAILS -----	43
APPENDIX B:	SOFTWARE DETAILS -----	47
APPENDIX C:	PACED DATA ACQUISITION USERS MANUAL -----	61
LIST OF REFERENCES	-----	68
INITIAL DISTRIBUTION LIST	-----	69

LIST OF TABLES

I.	CALL EXEC Parameters -----	29
II.	Data Acquisition Times -----	30
III.	Components used in PACER -----	30
C-1	Paced Data Output from Program A2D -----	65

LIST OF FIGURES

1.	Paced Data Acquisition System Components -----	31
2.	Schematic of PACER -----	32
3.	Original and Revised PACER Circuits -----	33
4.	CD4046 PLL Circuit Detail -----	34
5.	EXEC CALL Flow Diagram -----	35
6.	PACER Test Chassis -----	36
7.	PACER Front Panel -----	37
8.	Pulse Trains at Counter B1 for Analog & Digital PLL Circuits -----	38
9.	Paced I/O Request Flow Diagram -----	39
10.	Data Acquisition and Test Equipment -----	40
11.	Ramp Test Data from Original PACER -----	41
12.	Ramp Test Data from Revised PACER -----	42
C1	Data Acquisition and Test Equipment -----	66
C2	Cable Connections for Test Data Acquisition -----	67

LIST OF SYMBOLS AND ABBREVIATIONS

SYMBOLS

A	Driver Amplifier
B	4-Bit binary counter
C	Comparator
F	Buffer Amplifier
U	AND gate
L	Latching Flip Flop
D	Delay Flip Flop
I	Inverter
17 ₈	PACER I/O controller port
11 ₈	A/D I/O controller port

ABBREVIATIONS

A/D	Analog-to-Digital
I/O	Input-Output
RTE	Real-Time Executive
1/Rev	Once per Revolution
1/BL	Once per Blade Passage
PLL	Phase Lock Loop
CMOS	Complementary Metal Oxide Semiconductor
TTL	Transistor-Transistor Logic
DMA	Direct Memory Access
DCPC	Dual Channel Port Controller
TP	Test Point

ACKNOWLEDGEMENT

To my thesis advisor, Dr. Raymond P. Shreeve, goes my sincere appreciation for his steadfast moral and technical support. Without his timely and intuitive guidance, this project would not have culminated in the results reported herein.

Secondly, to Mr. Jack King goes credit for the fine workmanship resulting in the circuits built and tested for this report. His expertise in the field of electronics was an invaluable aid in the completion of this work.

Finally, to Mr. Alan McGuire goes my gratitude for his most professional work in the drawings and figures completed for this report.

I. INTRODUCTION

The device described herein and referred to as the "PACER" is part of a computer controlled data acquisition system in use at the Turbopropulsion Laboratory at the U.S. Naval Postgraduate School. It is an electronic interface unit built of solid state and integrated circuit components. The PACER was designed to allow the acquisition of data from high response transducers mounted in the case of rotating machines to be synchronized with respect to rotor position. Using the PACER, the analog to digital conversion of the data from a particular transducer can be programmed to occur at any position of the rotor with respect to the transducer, independent of rotor speed.

The PACER was first designed and built in a "bread board" configuration in 1976 by James C. West as described in Reference (1). U.S. patent no. 4,181,962 was issued for the PACER on January 1, 1980. The present hardware configuration of PACER involves minor but important changes which improve its performance and are documented in this report.

The original PACER made it difficult for the typical user to acquire accurate data in a reasonable amount of time for the following three reasons:

- (1) The timing pulses generated within PACER were not always spaced linearly in time between blade pair

synchronizing pulses. This resulted in data which in some cases was subtly distorted, and in other cases appeared to have noise riding on it.

- (2) The range over which the PACER could follow rotor RPM changes and remain synchronized was limited to approximately $\pm 15\%$ of the initial RPM at which the PACER was set to take the data. This required repeated, and somewhat involved, manual adjustment of an RPM "lock-on" procedure to acquire data at different speeds.
- (3) The rate at which data could be taken was limited below the desired rate. This meant that rather than being able to sample data on every revolution of the rotor, the system was only capable of taking data once every 8 to 10 revolutions, depending on RPM.

The methods used to improve the performance of the PACER fall into two areas, hardware and software. Hardware changes were used to improve PACER linearity, eliminate manual lock-on procedures, and increase speed-following range. A change in acquisition software was used to increase the rate at which data could be taken.

The change in PACER hardware consisted of replacing the original analog 562 phase lock loop with a CMOS digital phase lock loop and eliminating the discrete components forming the coupling circuit in the PLL feedback path. The change to acquisition software involved use of the DMA (direct memory

access) feature of RTE-IVB system software [Ref. 2] which is incorporated in the I/O driver written for the PACER.

As a result of the hardware and software changes which were made, all of the limitations described above were eliminated. The improvement in PACER performance was verified using test programs and rotating machine signal simulation circuits which enabled controlled test techniques to be employed.

In the following section of this report a description of the entire paced data acquisition system is given. Section III describes the changes made to PACER hardware and the effects of those changes, while Section IV describes the change to acquisition software. In Section V the results of the changes are verified with a report of the system tests. Section VI lists conclusions and recommendations for further system development. Appendix A contains detailed hardware circuit design figures and Appendix B details the software programs - both acquisition FORTRAN and system assembly language drivers. Finally Appendix C is a step by step system users manual for paced data acquisition.

II. PACED DATA ACQUISITION SYSTEM DESCRIPTION

A. GENERAL DESCRIPTION

Components of the system are shown schematically in Figure 1 and details of the circuits, including the modifications made in the present work are shown in Figures 2-4. The PACER acts as a secondary controller on the interface between the Hewlett-Packard HP 21-MX computer and the A/D converter. Referring to Fig. 1, in a normal (not paced) data taking sequence, the 21MX would call on the A/D converter to take an analog data sample, convert it to digital, and output it to the computer memory. Since the computer program execution cannot be synchronized to the rotation of the machine shaft, the data sample would be from a random, unidentified point.

In a paced data acquisition sequence, the PACER provides the timing control to the 21MX computer. After the 21 MX computer passes a word (IBLADE) to the PACER defining the desired position, the PACER acts as an intermediary. It intercepts the computer command to the A/D converter, tells the computer that the A/D converter is in the process of acquiring the data, then sends a command pulse to the A/D converter at a time synchronized to the desired position in the cycle of the rotation of the machine.

The sequence of events for paced data acquisition using the software developed in the present work is as follows:

- (1) The user enters the main program (which was written to be used for system testing or for data acquisition).
- (2) The main program prompts the user for information regarding the (rotor) position(s) desired at which to start taking data points. This information defines the integer IBLADE.
- (3) The main program calls the PACER, passes IBLADE to the PACER, and receives rotation speed (IRPM) from the PACER. Control then returns to the main program.
- (4) The main program calls the A/D converter telling it to take a number of data samples (N) (at the desired point). When complete, control is returned to the main program.
- (5) If a survey of positions (for example, across a pair of blade passages) has been programmed using a DO loop, the main program repeats steps (2) and (3), incrementing IBLADE each time until the loop is finished.
- (6) When all data have been taken and stored in the computer memory, the main program converts the digital data (which are binary whole numbers) to decimal values scaled appropriately to the ± 1.0 volt range of the A/D converter. As programmed, it then outputs that data to the desired peripheral(s) (i.e. the printer, plotter, or terminal).

B. SYSTEM SOFTWARE

The software used in the data acquisition should be viewed as consisting of two separate parts, the RTE-IVB operating system which is generated in-house following standard procedures supplied by Hewlett-Packard, and the system test and operation FORTRAN program which may be modified at any time by the user operating in the RTE-IVB system.

1. RTE-IVB Operating System

The RTE-IVB (Software) Operating System is generated (and can be regenerated) by the System Manager in a process which "configures" the System for the particular set of (I/O) devices which the computer must address [Ref. 3]. RTE-IVB permanently resides on disc and is automatically loaded when the system is turned on. It consists of a collection of software modules which perform system resource management, operator requests for utility programs (FORTRAN compiler, file editor, etc.), and user program scheduling for time sharing [Ref. 4]. RTE-IVB is visible to the user through interaction at the terminal. It allows multi-programming through its scheduling modules so that more than one user's program may be active at a time. The input/output (I/O) drivers are a set of modules in the RTE-IVB System. They are the software routines which control the input and output communication between the user's program and addressed peripheral devices. The drivers enable efficient use of peripherals which act at different speeds by allowing one or more fast I/O requests to be

processed while waiting for a request from a slow peripheral device to be completed. A driver written for the PACER (DVR.70) and a driver written for the A/D converter (DVR.56) are part of RTE-IVB and are listed in Appendix B.

2. System Test and Operation Program

The system test and operation program (A2D) is a FORTRAN program written and used in the course of the present work. A listing and flow diagram are given in Appendix B. Program A2D converts the user's requests, which are entered at the terminal, to the parameters required by the RTE-IVB I/O drivers. It is an interactive program consisting of two parts. The first part, a system test (subroutine ADTES), is entered if the user wants to carry out a test of the paced data acquisition system simply to ensure that all components are operating correctly. The second part, (Subroutine RPACE), is executed if paced data is to be acquired from a test rig. Both the "test" and "operation" portions of A2D use the FORTRAN statement "CALL EXEC" to enter the appropriate driver. The CALL EXEC statement, with its accompanying parameters, transfers control from the FORTRAN program to the assembly language driver for the device requested. A simplified flow diagram of the CALL EXEC routine is shown in Figure 5. The driver initiates the input or output task as specified in the parameters which it received. If the task is for "output", after the task is initiated control may return to the calling FORTRAN program or another user's program. If the task

requires "input", then control may be passed to another program, but not back to the calling program, since the calling program must have an input value to continue executing. This permits efficient use of the computer's time, which is essential for multi-programming, while waiting for a slow peripheral device to complete its cycle of operation.

C. SYSTEM HARDWARE

The hardware devices used in paced data acquisition are the HP-21MX computer with printer, its magnetic disc, plotter, and terminal, the HP 5610A A/D converter and the PACER.

1. Hewlett Packard HP 21 MX Computer

The HP 21 MX is a (Micro-programmable) mini-computer having 128 machine instructions and 32K of logical main frame memory. In the present configuration a 20 megabyte capacity disc and disc operating system are an integral part of the system. A detailed description of the computer is given in Reference 2.

An important feature which is typical of computers of this size is the input-output structure. With a limited number of relatively slow I/O devices to be serviced, the computer can communicate with all devices through a single port known as the I/O bus. Each device requires its own I/O interface on the bus. The interface acts as a filter and ensures that output information is received only by the device designated to receive it and that input information is put

on the bus from only one device at a time. The I/O software drivers control the I/O hardware interfaces by commands to either "turn-on" or "turn-off".

2. Hewlett Packard HP 5610A A/D Converter

The HP 5610A analog-to-digital converter accepts analog data input on up to sixteen different channels and under computer controlled multiplexing converts to a 10 bit binary data output. With an input conversion aperture of 50 nanoseconds, rapidly changing signals (100 KHz) can be converted accurately. The HP 5610A can operate in one of six modes as described in Reference 5. Currently the paced data acquisition system uses the "random access mode" in which a specific channel is sampled on receipt of a command word and an encode command pulse from the 21 MX computer. The command word tells the A/D converter which mode of operation to use and which channel number to sample. The encode command pulse triggers the data conversion to start 2 μ sec later. The data conversion itself is finished in a total time of 10 μ sec. Using computer-issued encodes, which is the mode required for paced data, the sample cycle time is 20 μ sec. Hence data can be converted at rates of up to 50,000 samples per seconds, depending on how rapidly each successive command word is received.

The other mode which is used only for non-paced data is the Free Run, Random Access mode. In this mode the command word is required as before, but no encode command is

needed from the computer. The A/D converter simply converts data as fast as it can (100,000 samples per second) on the selected channel. This mode is not addressed further in this report.

3. PACER

A schematic of the PACER is shown in Figure 2. In its original form, a detailed description of the internal operation is given in Ref. 1. The PACER consists of two major sections, an "RPM counting section" and a "synchronized command pulse section". The "RPM counting section" continuously counts the number of 250 KHz time base pulses that occur between the once-per-revolution pulses received from the test rig. This number of counts is available as an output (IRPM) from the PACER on every revolution cycle.

The "synchronized command pulse" section is the heart of the PACER. It uses a phase lock loop to generate 256 pulses within each pair of blade passages (i.e. 128 pulses from blade #1 to blade #2 and 128 pulses from blade #2 to blade #3). At the same time, these pulses are counted and compared with the programmed data conversion location specified in IBLADE. When the comparison is true, a command to the A/D converter (A/D Device Command) is generated. Thus a command to convert a data sample is synchronized with a desired position of the rotation rotor in the machine.

III. CHANGES TO PACER HARDWARE

In order to determine the cause of the non-linearity in the PACER, a test chassis was built to provide easy access to the four circuit boards and to allow modifications to be attempted without interference to the working unit. The test chassis is shown in Figure 6. It is electrically identical to the system PACER shown in Figure 7 and uses the same four circuit boards. Using the test PACER with an oscilloscope it was possible to examine the wave forms, at any point in the PACER circuit. In so doing, it was found that even with the lock-on procedure recommended in Reference 1, the output pulses from the PLL ($256 \cdot F_o/2$) were not always linearly spaced between the beginning and end of the input pulses ($F_o/2$). This non-linearity is seen in the oscilloscope traces shown in Figure 8, which shows the signal at counter B1. At counter B1 the pulse frequency is $1/32$ of the output frequency of the PLL which allows the non-linearity to be obvious to the eye. It was further noted that a deviation of as little as 3° from the ideal 270° phase relation called for in Reference 1, caused non-linear spacing and excessive unsteadiness ('jitter') of the pulses into counter B1. These problems were inherent in the 562N PLL when used with digital waveforms because an analog phase comparator was used in that particular circuit [Ref. 6].

A CD4046 (CMOS) PLL was therefore chosen to replace the 562N. The CD4046 uses a digital phase comparator to maintain lock [Ref. 7] and is specifically designed to operate with digital waveform inputs as are found in the PACER application. It also permits, with proper associated component design, operation over an extremely wide frequency range (by so-called frequency tracking) without losing lock.

The changes which were made in the PLL and associated circuitry are shown in Figure 3. Both the PLL and the discrete component coupling circuits were changed. The replacement of the old coupling circuits with CMOS-to-TTL (4050B Buffer) and TTL-to-CMOS (7417 Drivers with pull-up resistors) matching devices was necessary because of the special requirements of the CMOS PLL with regard to interfacing [Ref. 7]. The detailed circuitry of the CD4046 (CMOS) PLL is shown in Figure 4. Specific details of the components are given in Appendix A.

IV. CHANGE TO ACQUISITION SOFTWARE

A. METHODS OF INPUT/OUTPUT

The two methods available under RTE-IVB for input and output are the "standard" method and Direct Memory Access (DMA). In both methods the software driver controls the initiation and completion of the I/O request. Figure 9 is a schematic representation of the hardware and software involved in an I/O request in the paced data acquisition process. The standard I/O method requires that the software driver be entered for each data sample taken. In contrast, the DMA I/O method uses the "dual channel port controller" option of the 21 MX computer to bypass the requirement to return to the driver for each new data sample [Ref. 2]. Thus by using DMA, the time involved in executing the software driver for each sample is saved.

B. INCORPORATION OF DMA

The system software was changed so that DMA was used for the A/D I/O process. The DCPC option was added to the system in 1977. The driver DVR56 was subsequently modified by Hewlett Packard to permit DMA for I/O operation with the A/D converter. The use of the DMA feature required only that the proper parameters be specified in the CALL EXEC statement for the A/D converter. Table I lists the parameters,

with their meanings, for the CALL EXEC statements used to call the A/D converter and the PACER through the drivers DVR56 and DVR70 respectively. The parameter "N", which is passed in the call to driver DVR56, sets up the DMA option in the 21 MX I/O interface logic through the Dual Channel Port Controller (DCPC). The program A2D was written so as to use the DMA feature. A flow chart, listings, and parameters used in program A2D and the drivers DVR56 and DVR70 are given in Appendix B.

V. RESULTS

Tests were run to verify the linearity of the new CMOS PLL circuitry, to demonstrate the automatic lock on feature, and to determine the speed at which data was acquired. The tests were run using the test pulse generation circuit on circuit board #4 of the PACER. This circuit provides an electronically produced simulation of the 1/Rev and 1/Blade pulses that would ordinarily be received from the test rig. The test set up for the tests is shown in Figure 10. An external signal generator was used to provide the driving signal to the pulse generating circuit at the desired blade passing frequency. Appendix C gives detailed procedures for performing a simulation test run.

A. LINEARITY TEST

Figure 8 shows a comparison of PLL output pulses from the 562N PLL and the new CMOS digital PLL circuits. It can be seen that the new circuitry produces symmetric and evenly spaced pulses while the old PLL circuit does not.

A linear ramp test signal was input to the A/D converter on analog channel 0. The PACER test portion of program A2D was run calling for a survey across the simulated blade pair. The test was repeated for the old and new PLL circuits. Figures 11 and 12 show the output results from the PACER

using the old and new PLL circuits respectively. The apparent "bending" of the ramp test signal when seen as the graphed output from the old PLL method is due to the inherent non-linearity of the 562N PLL. The strict linearity of the CMOS digital PLL circuit was noted.

B. AUTO LOCK-ON TEST

The new CMOS digital PLL requires no lock-on procedures as did the 562N PLL [Ref. 1]. Tests were run to confirm that while varying the blade passing frequency, the new PLL remained in a locked-on condition. It was shown that within the design range of the PLL circuitry, any variation of blade passing frequency (RPM) was followed without error by the digital phase lock loop. Two separate PLL circuits were designed, each one covering a range of blade passing frequencies. One PLL circuit now covers the range from 250 Hz to 2.5 KHz. The other covers the higher range from 3 KHz to 11.1 KHz. The reasons for this division are explained in Appendix A.

C. TEST OF ACQUISITION TIME

Using the software methods used in Reference 1, a short test program calling for a specified number of data samples to be taken, was run. Clock time accurate to .1 millisec was recorded by the program just before the first sample and just after the last sample of data was acquired. The lapsed time for the total acquisition was output. It was shown that up to 10 revolutions of the machine rotor were required for each data sample to be taken.

After changing to the DMA software method described in section IV, similar tests were run. The results of these tests are shown in Table II. It was noted that the interval between samples was reduced to less than one revolution of the machine rotor.

VI. CONCLUSIONS AND RECOMMENDATIONS

The desired improvements in the paced data acquisition system were achieved; namely,

- (1) The speed of acquisition of successive data samples was increased to enable data to be sampled on every revolution.
- (2) The correlation between the position recorded for a paced data sample and the physical position of the probe with respect to the rotor at acquisition, was significantly improved through an improvement in the linearity and stability of the PLL and associated circuitry.
- (3) The manual adjustments previously required for each small range of RPM were entirely eliminated by the reported hardware modifications.

With the present hardware and software the PACER operates as fast as is possible given the constraint that the 21 MX computer operates always in the interrupt mode for all I/O operations. If the need arises to survey across a blade pair on one resolution and the computer can be dedicated to the single task of acquiring paced data, then the non-interrupt mode of 21 MX I/O processing could be used. This change would eliminate other users during the paced data program operation. It would require that the drivers DVR56 and DVR70

to be rewritten in assembly language and loaded into the RTE-IVB operating system by the system manager. It is noted however that the maximum data rate of 100,000 samples per sec cannot be exceeded using the present A/D converter.

Table I. CALL EXEC Parameters

To call the PACER (DVR70)

CALL EXEC (1, LU, IRPM, LEN, IBLADE)

<u>Parameter</u>	<u>Meaning</u>	<u>Limits/Value</u>
1	I/O	1
LU	device reference number	19
IRPM	RPM timing counts returned	N/A
LEN	number IRPM of words passed	0,1
IBLADE	data position indicator	0-35,584

To clear the PACER

CALL EXEC (3, LU)

<u>Parameter</u>	<u>Meaning</u>	<u>Limits/Value</u>
3	clear the device	3
LU	as above	19

To call the A/D (DVR56)

CALL EXEC (1, IDRT, IBUF, N, ICHAN, ICODE)

<u>Parameter</u>	<u>Meaning</u>	<u>Limits/Value</u>
1	I/O	1
IDRT	device reference number	20
IBUFF	data storage array name	dimension 256
N	number of samples	1-99
ICHAN	input channel number	0-15
ICODE	mode of A/D operation	0-7

Table II. Data Acquisition Times

Run	Number Samples	RPM	Time	Time/Rev	Time/Sample
<u>Before DMA</u>					
1	100	17,300	1.61	.0035 sec.	.0161 sec.
2	100	17,400	1.60	.0032 "	.016 "
3	20	7,500	.51	.008 "	.025 "
4	20	8,000	.45	.0075 "	.0225 "
5	500	30,000	9.51	.002 "	.019 "
6	500	29,900	9.50	.002 "	.019 "
<u>After DMA</u>					
1	100	15,100	.398	.00397	.00398
2	100	15,000	.400	.004	.004
3	100	8,000	.750	.0075	.0075
4	100	30,000	.200	.002	.002

Table III. Components Used in PACER

COMPONENT	SCHEMATIC NUMBER	VALUE OR TYPE NO.	
		High Board	Low Board
Resistors	R1	10 K Ω	4.7 K Ω
	R2	100 K Ω	100 K Ω
	R3	1 M Ω	1 M Ω
	R4	39 K Ω	47 K Ω
	R5	12 K Ω	12 K Ω
	R6	12 K Ω	12 K Ω
	R7	10 K Ω	12 K Ω
Capacitors	C1	50 pf	.001 μ f
	C2	1.5 μ f	1.5 μ f
Counter	B1 thru B10	74193	
Latch	L1 thru L8	7475	
Comparator	C1 thru C4	9324	
AND Gate	U1 thru U3	7408	
Inverter	I1, I2	7404	
Buffer	F1 thru F5	N4050B	
Driver	A1 thru A4	7417N	
Phase Lock Loop	PLL	CD4046	

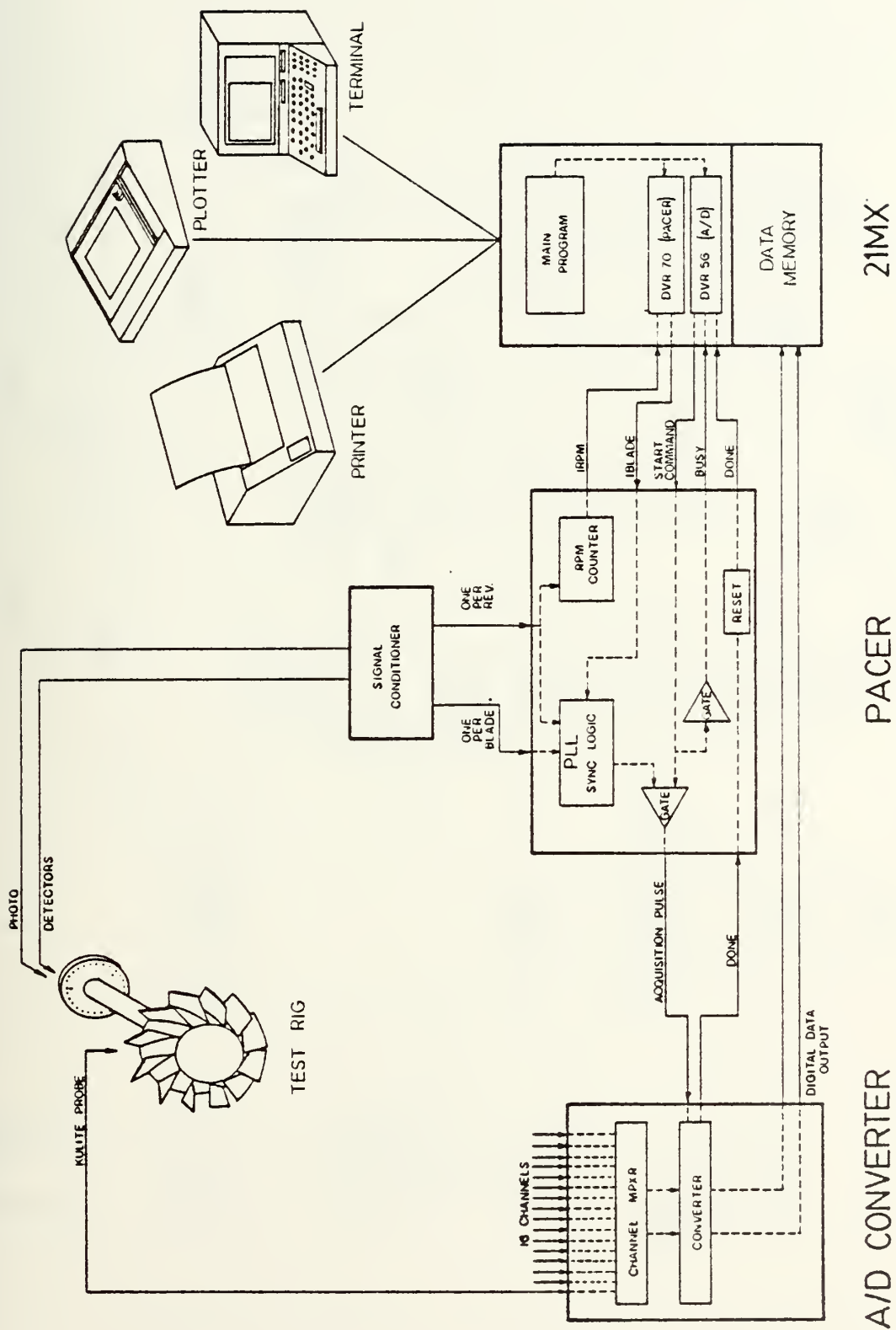


Figure 1. Paced Data Acquisition System Components

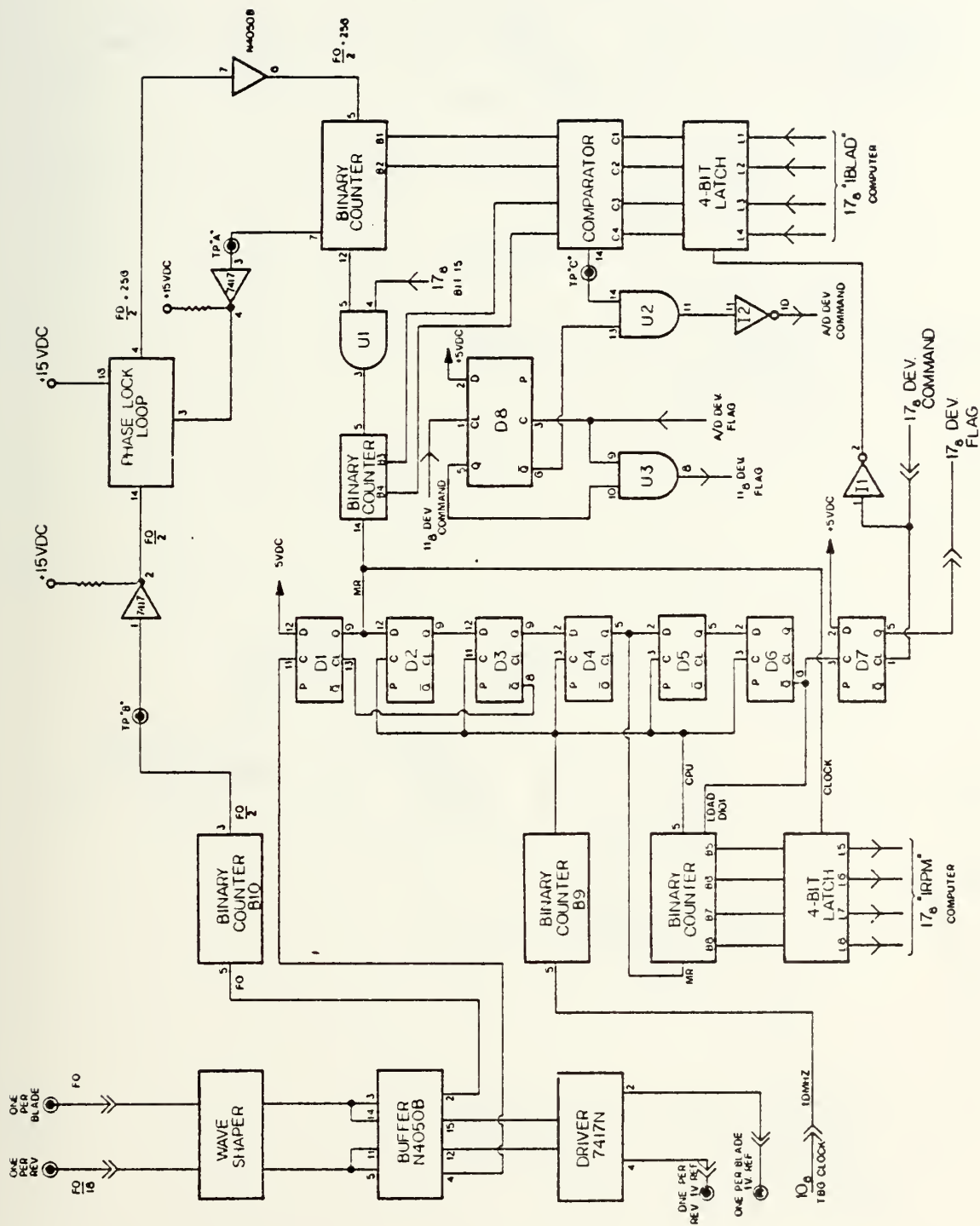


Figure 2. Schematic of PACER

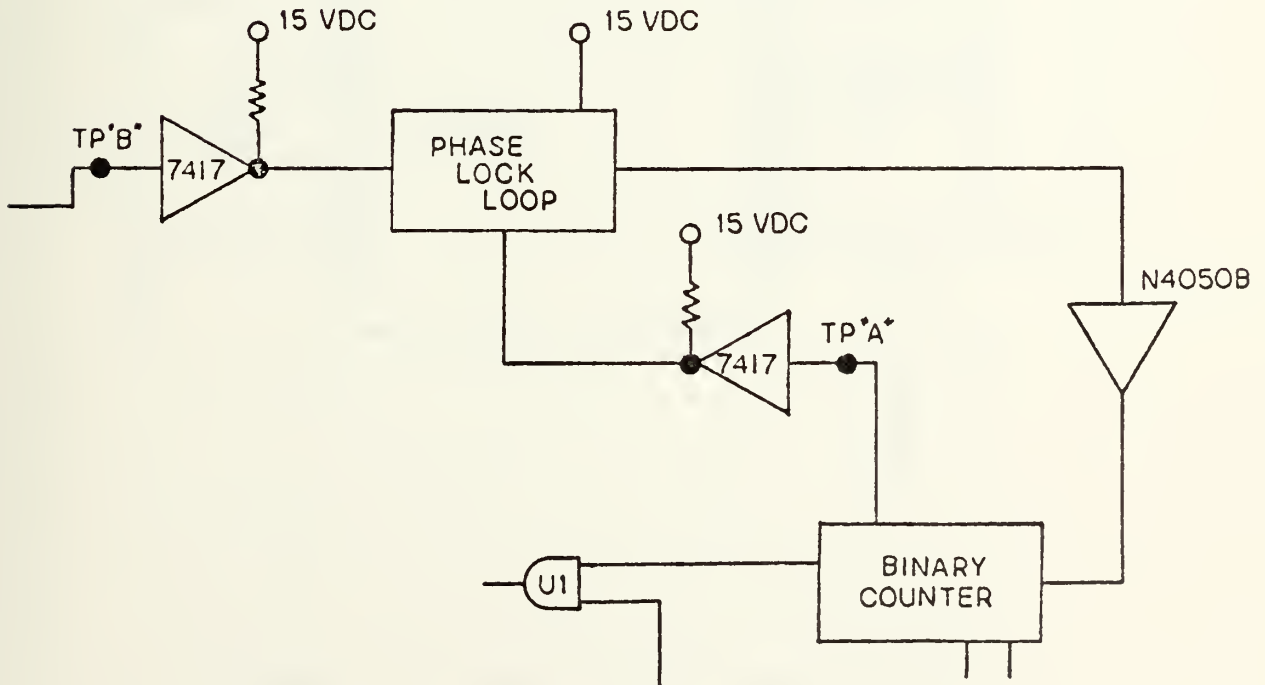
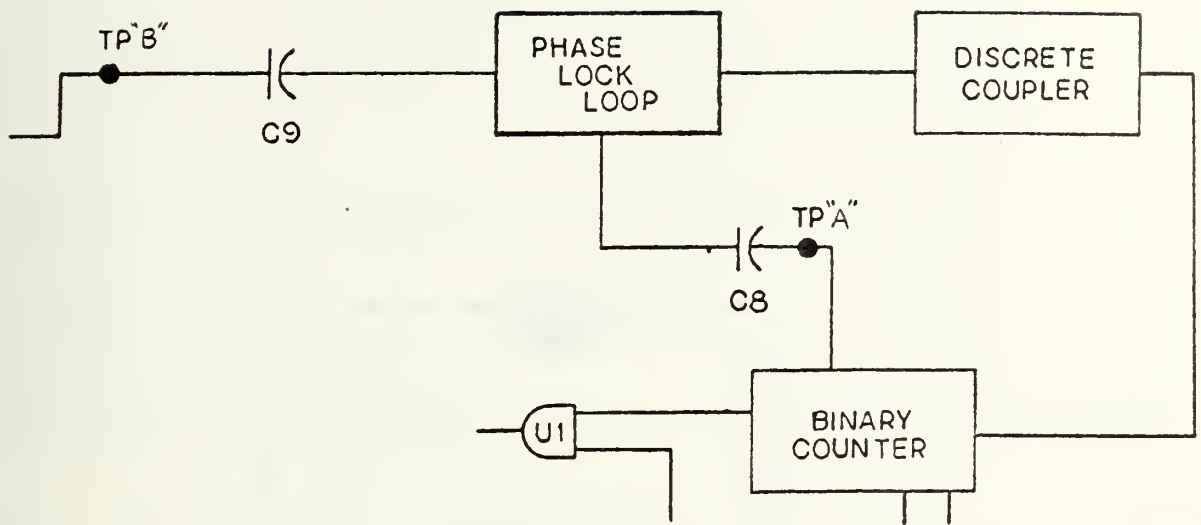


Figure 3. Original and Revised PACER Circuits

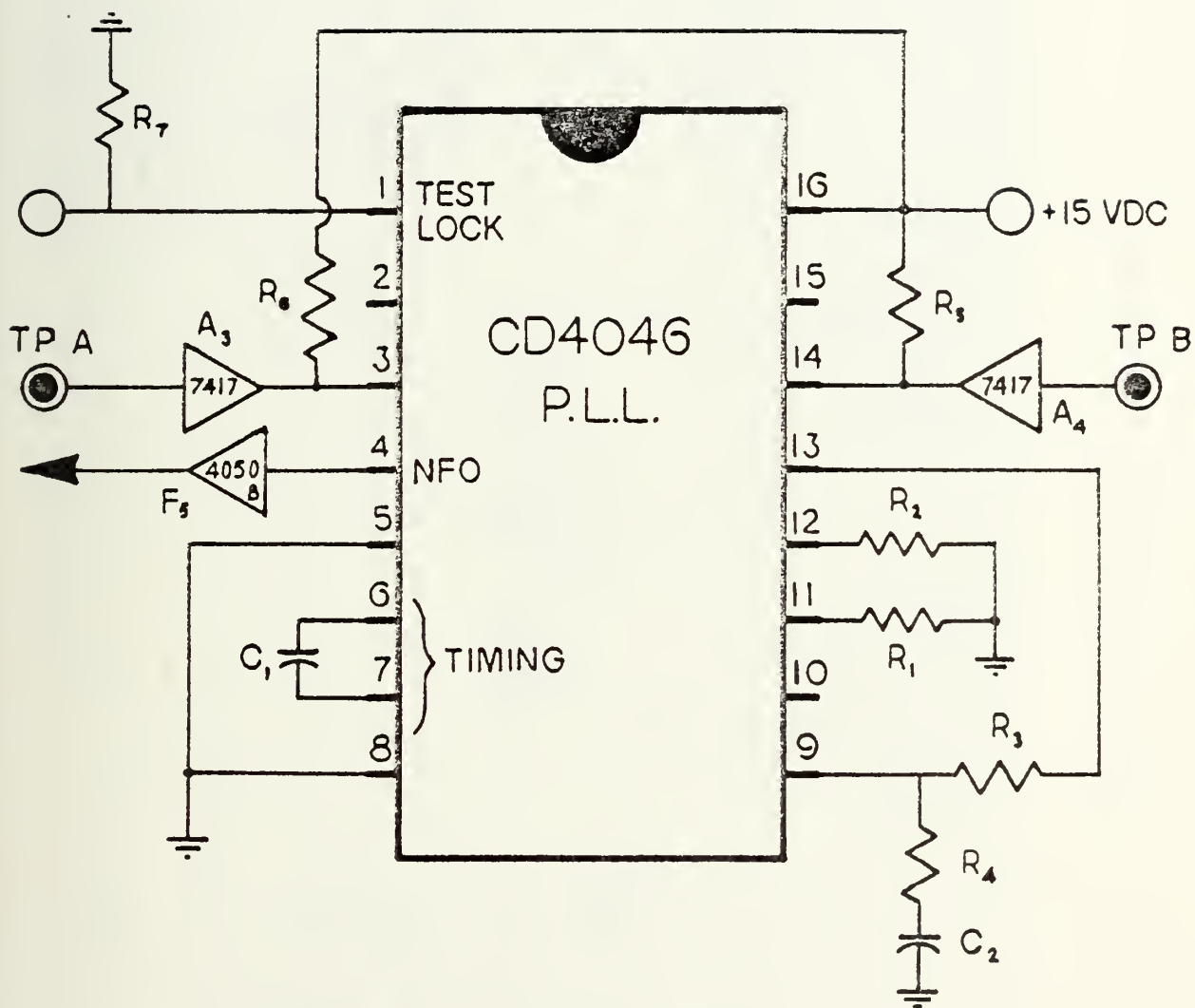


Figure 4. CD4046 PLL Circuit Detail

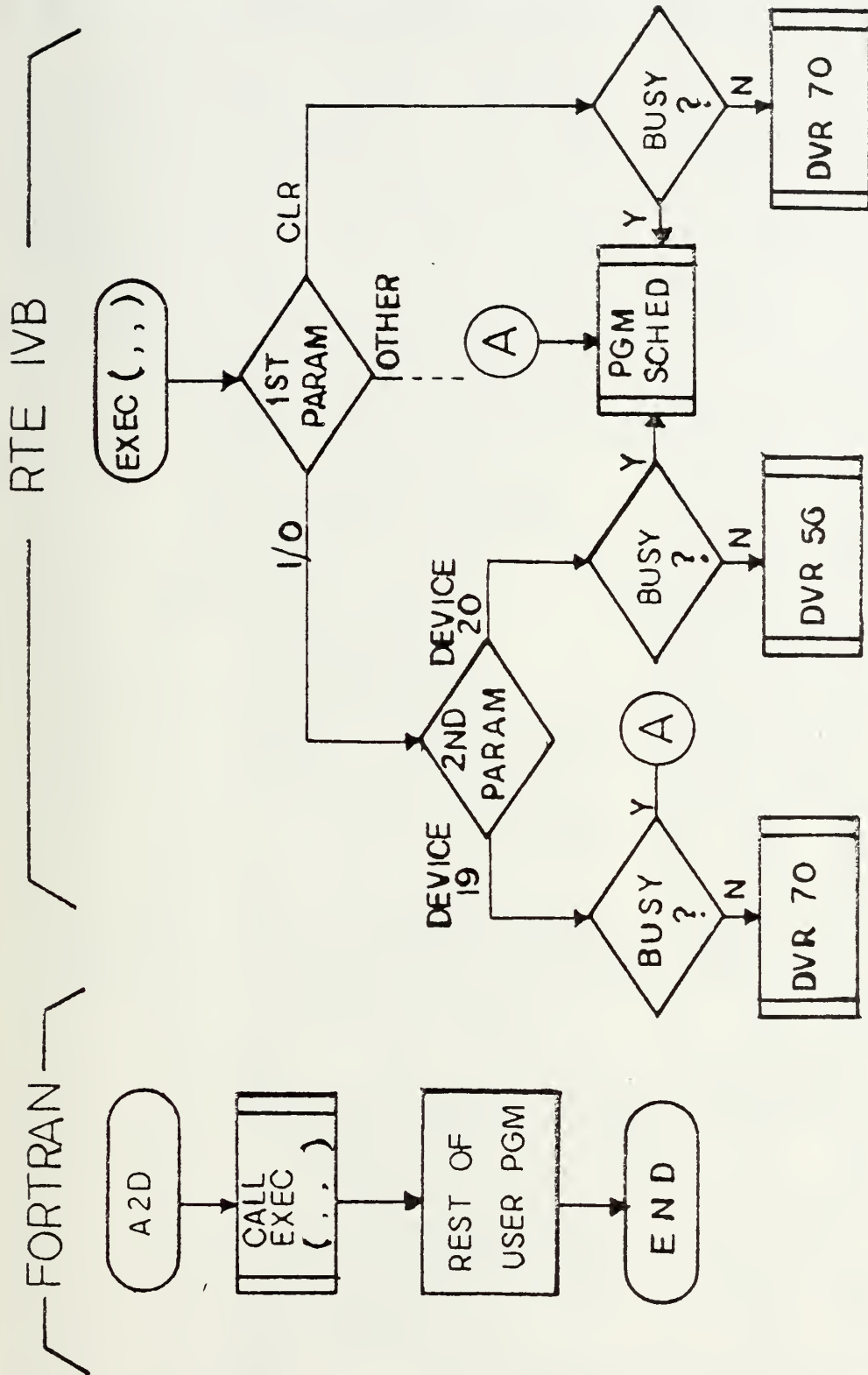


Figure 5. EXEC CALL Flow Diagram

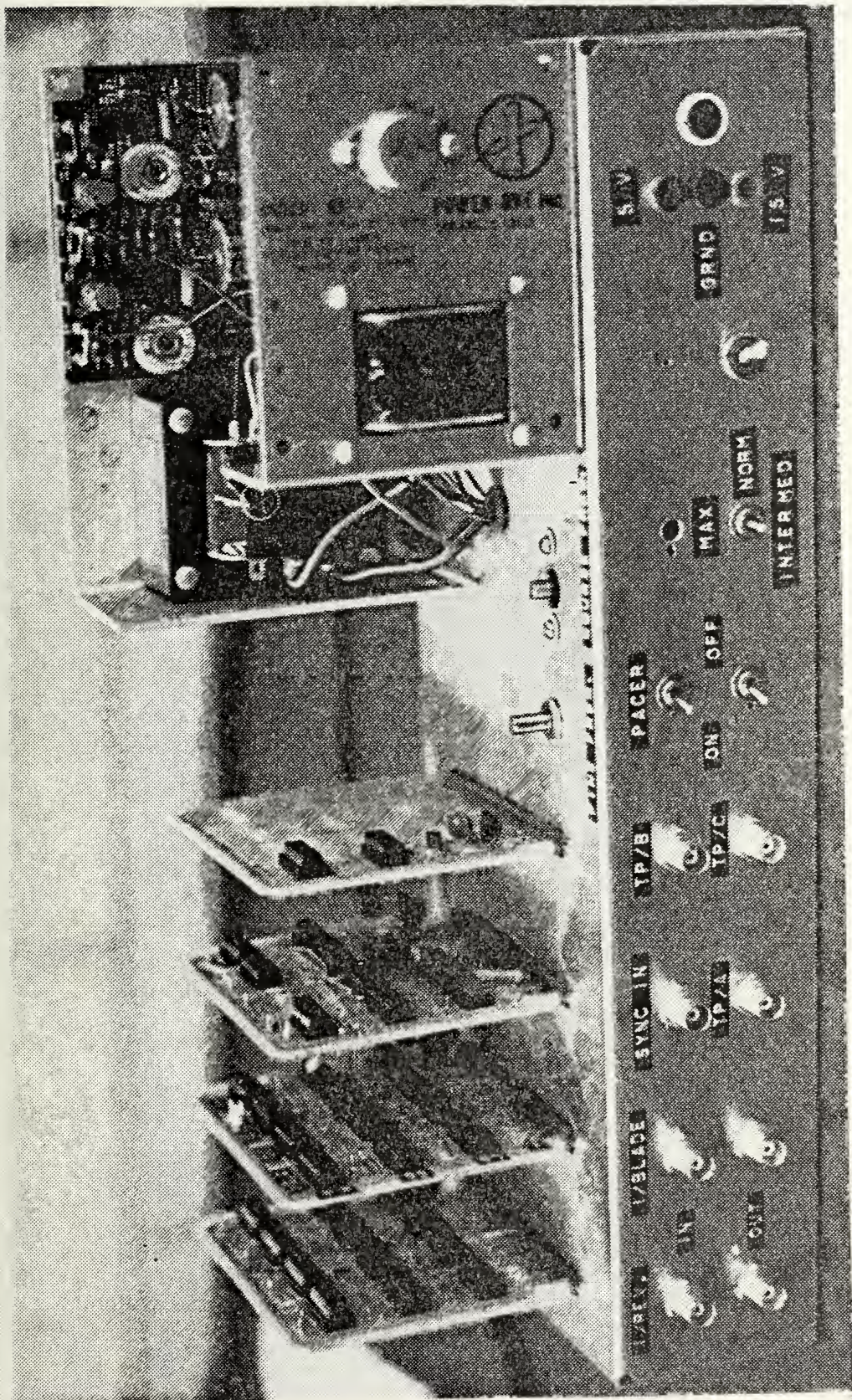


Figure 6. PACER Test Chassis

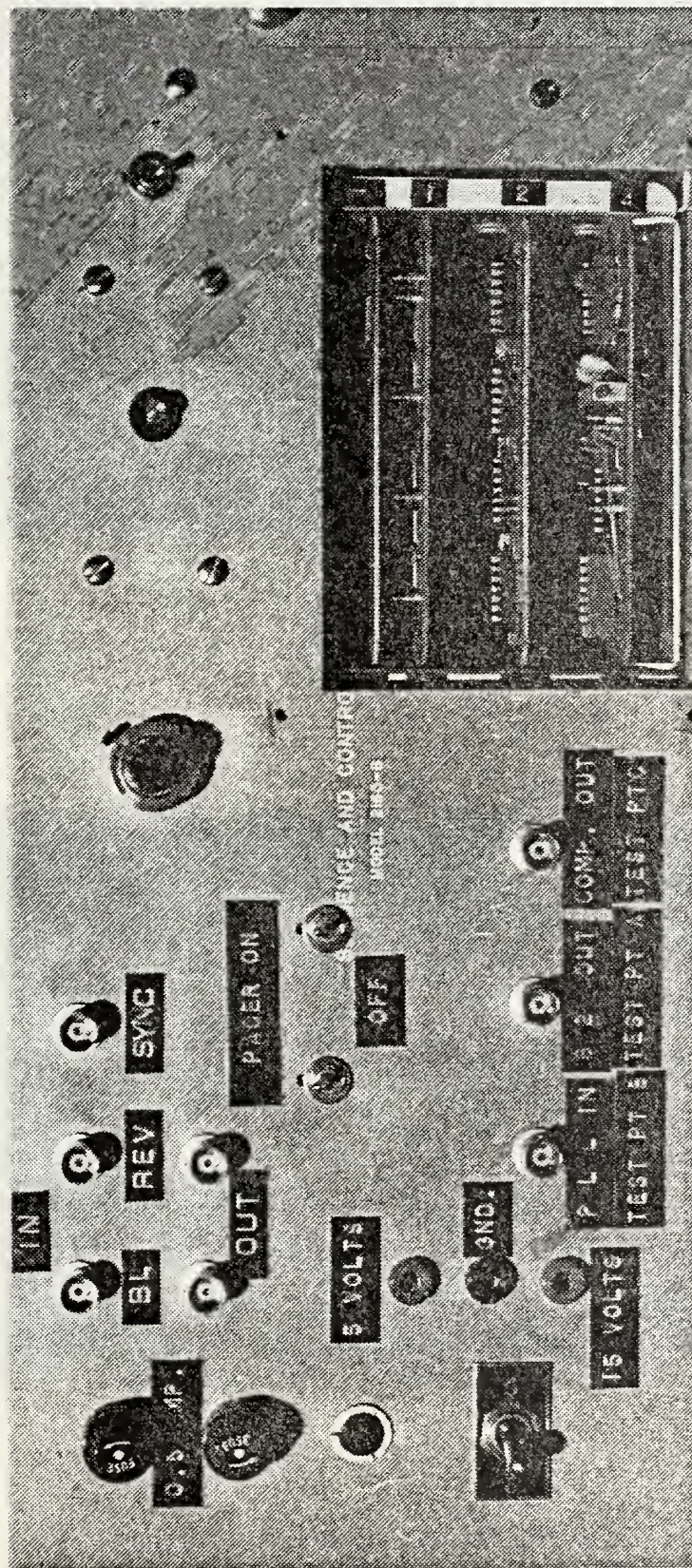
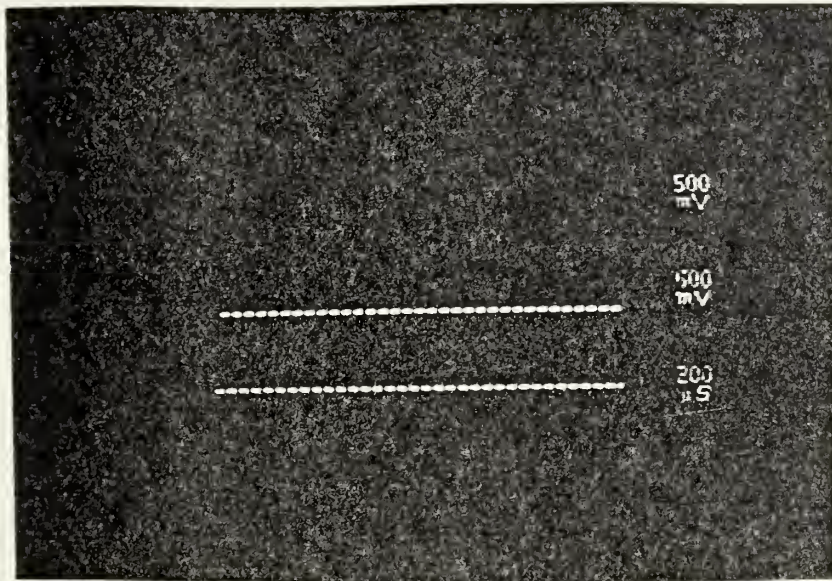
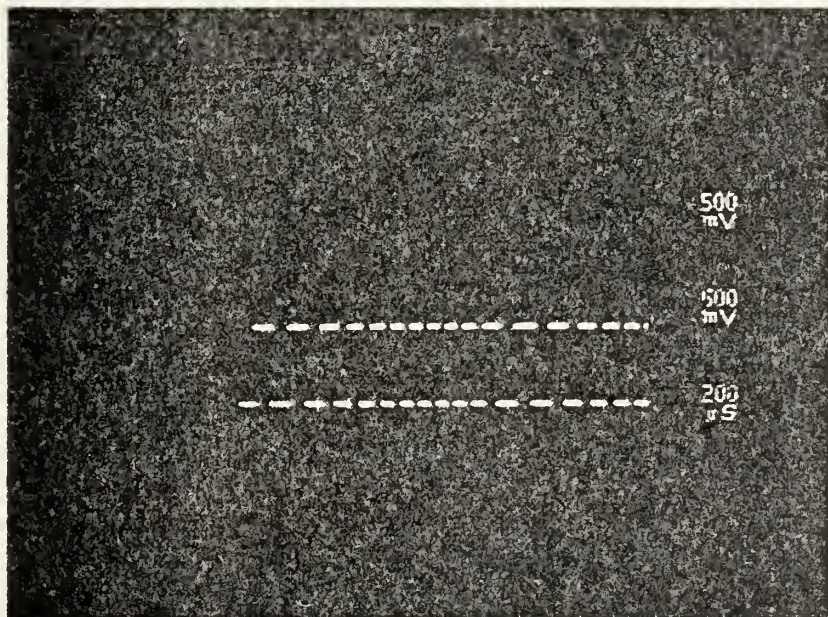


Figure 7. PACER Front Panel



(a) Digital PLL



(b) Analog PLL

Figure 8. Pulse Trains at Counter B1 for Analog & Digital PLL Circuits

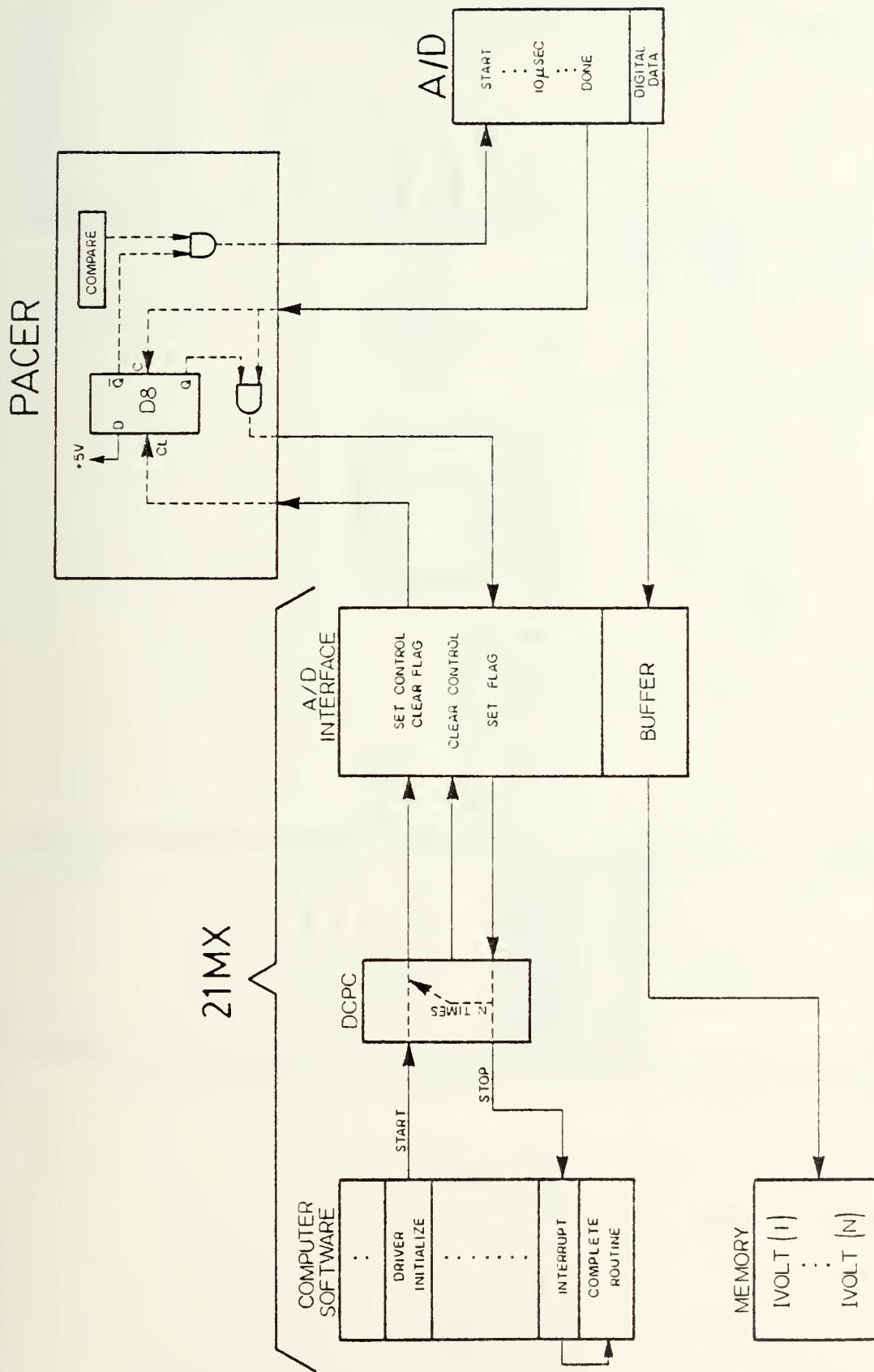


Figure 9. Paced I/O Request Flow Diagram

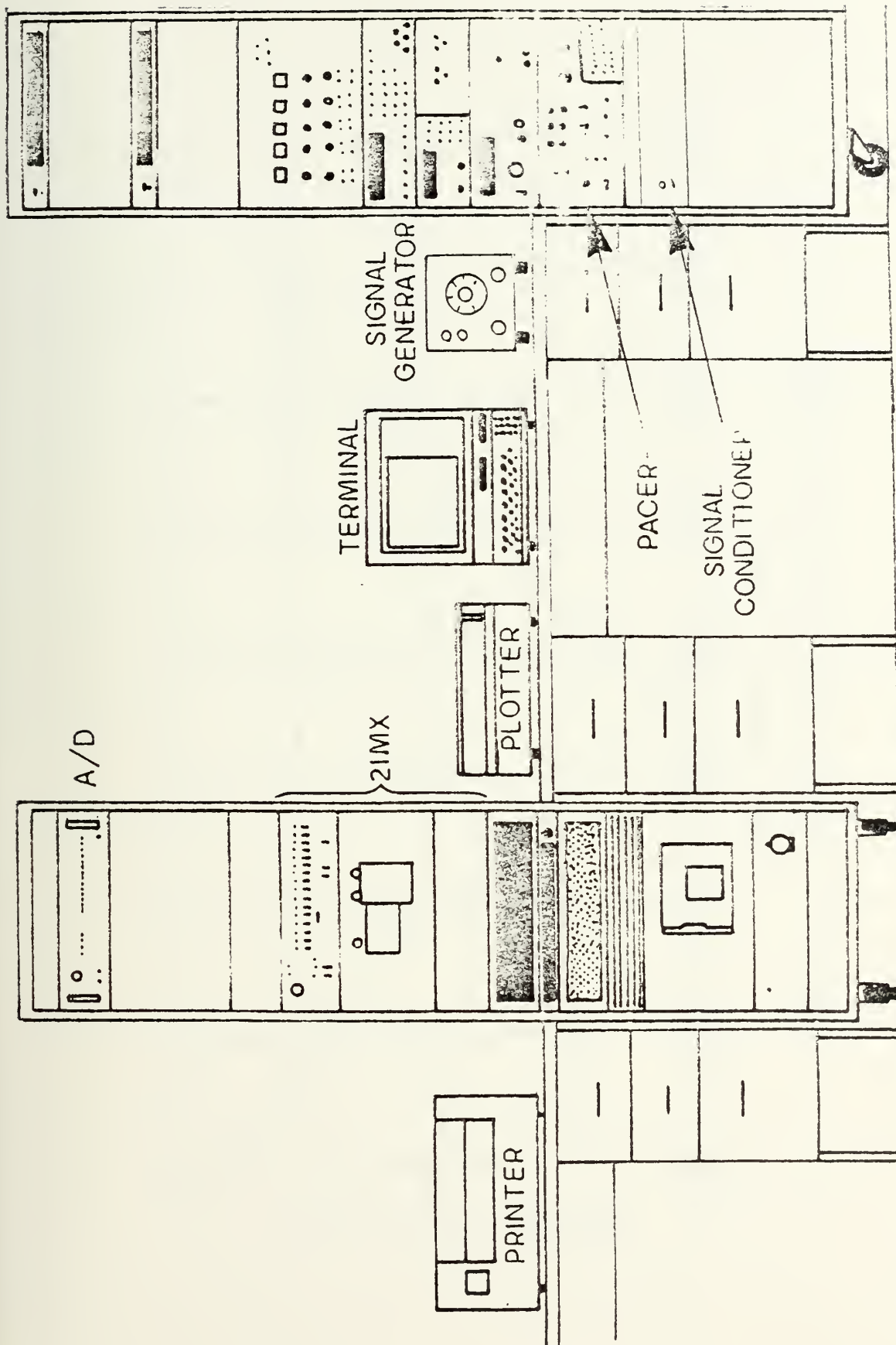


Figure 10. Data Acquisition and Test Equipment

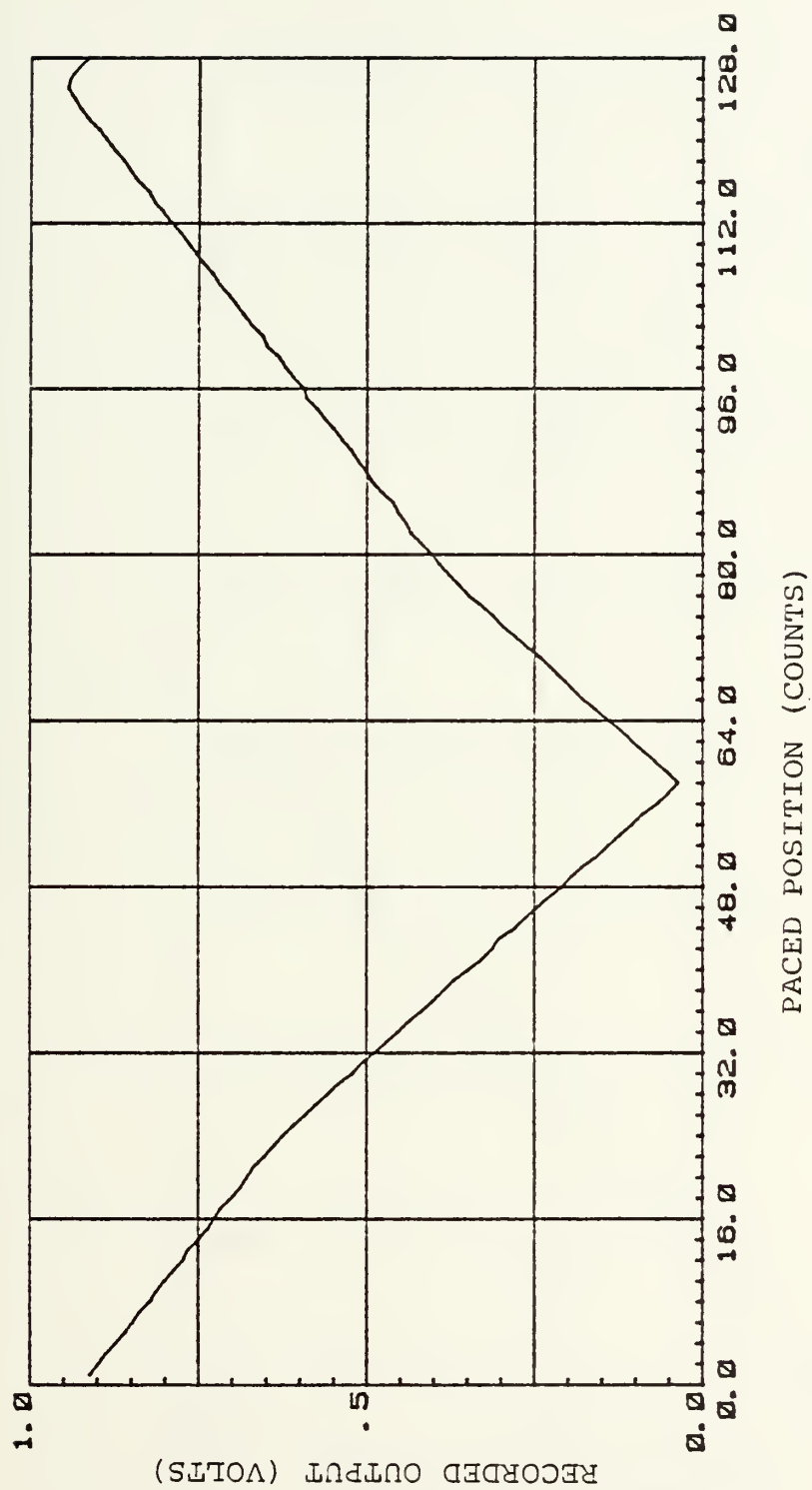
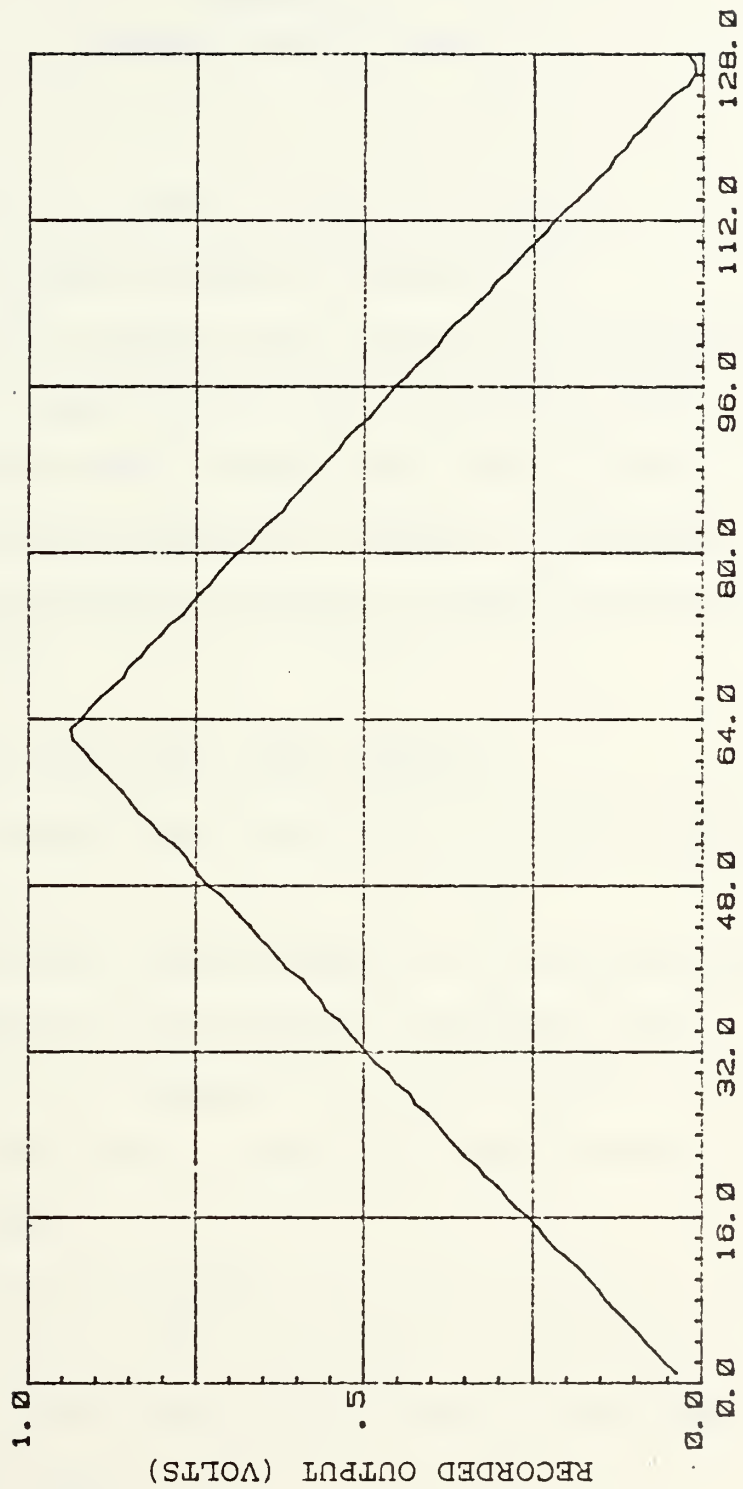


Figure 11. Ramp Test Data from Original PACER



PACED POSITION (COUNTS)

Figure 12. Ramp Test Data from Revised PACER

APPENDIX A

HARDWARE DESIGN DETAILS

A.1 INTRODUCTION

The PLL circuit is shown in detail in Figure 4. A listing of component values is found in Table III.

Two separate PLL circuits were designed and incorporated into the hardware; one for each of two frequency ranges. This was done in order to cover a very large total frequency range while maintaining fast response to changes in frequency [Ref. 7]. In the following sections the design procedure which was followed is documented.

A.2 DIGITAL PHASE LOCK LOOP (CMOS) DESIGN

The CD4046 digital PLL requires four areas of external design [Ref. 7].

- (1) Selecting the timing capacitor C_1 which determines the center of the operating frequency range.
- (2) Selecting the values of R_2 and ratio of R_1 to R_2 which determine the upper and lower bounds of the lock range.
- (3) Selecting the ratios of R_3 to R_4 , R_3 to C_2 , and their values, which contribute to determine the damping ratio and settling time of the second order feedback loop.

- (4) Interfacing the CMOS integrated circuit design with the TTL integrated circuits already in the PACER.

These areas are detailed in the following sections.

A.2.1 Timing Capacitor

In the following discussion, figures and pages are quoted with respect to Reference 8, the main source for design information. To begin the design a value of R_2 was chosen within the limits listed on page 228 of Ref. 8. The value of C_1 was approximated using figure 5(b) of Ref. 8. The value was then readjusted after testing to compensate for the effects of the following component values.

A.2.2 R_1/R_2

The chosen frequency range (f_{max}/f_{min}) was used to enter figure (c) of Ref. 8. The ratio R_1/R_2 was obtained from the data in that figure using the design value of the supply voltage to the PLL. Knowing the ratio R_1/R_2 and the value of R_2 selected in section A.2.1, the value of R_1 was obtained.

A.2.3 $R_3/R_4/C_2$

The design of the loop low-pass filter was a trial and error iterative process because of effects from the counting circuits B1 and B2 present in the loop [Ref. 7]. The RC time constant of R_3 and C_2 determined the settling time of the loop while the ratio of R_3 to R_4 determined the damping ratio.

The nominal values found in Reference 7 were used initially and then these were adjusted to obtain what was considered to be the best loop response to changes in the input frequency. Loop response time was found by putting small but rapid perturbations on the test frequency, then noting the time to regain phase lock-on. By balancing the response time (required to be as fast as possible) against the settling time resulting from the loop damping ratio (at a minimum to maintain stability) across the frequency range, a satisfactory overall loop response was attained.

A.2.4 Interfacing

Due to the extremely high input and output impedances of CMOS integrated circuits, an interfacing buffer was needed between the CMOS PLL output from pin 3 and the TTL counter (B1) input to pin 5 (Fig. 2). Also, interface drivers were needed between the outputs of TTL counters B10 and B2 and the inputs to the CMOS PLL at pins 14 and 3, respectively.

The buffer between PLL pin 4 and counter B1 pin 5 simply required wiring one of the unused buffers which were part of the N4050B Hex buffer chip already in the PACER. Since the N4050B used a +5 VDC supply, the required transition from PLL +15V logic level to the counter +5v logic level was made.

In order to transition from the TTL (+5v) logic level of counters B10 and B2 outputs to the required PLL input levels (greater than +7v for logic 1), two 7417N TTL drivers

were used with 12 K Ω "pull up" resistors on their outputs. This gave a high logic level of +15v and a low state current drain on the drivers of only 1.25 ma each, well within their fan out capability [Ref. 9].

APPENDIX B

SOFTWARE DETAILS

This Appendix contains the following materials:

B.1 ACQUISITION FORTRAN PROGRAM A2D [Ref. 10]

B.1.1 Program A2D Flow Chart

B.1.2 Program A2D Listing

B.1.3 Program A2D Parameter Listing

B.2 SOFTWARE DRIVERS [Ref. 11, See Note 1]

B.2.1 Flow Chart

B.2.2 Pacer Driver DVR 70

B.2.3 A/D Driver DVR 56

Notes on Software Drivers

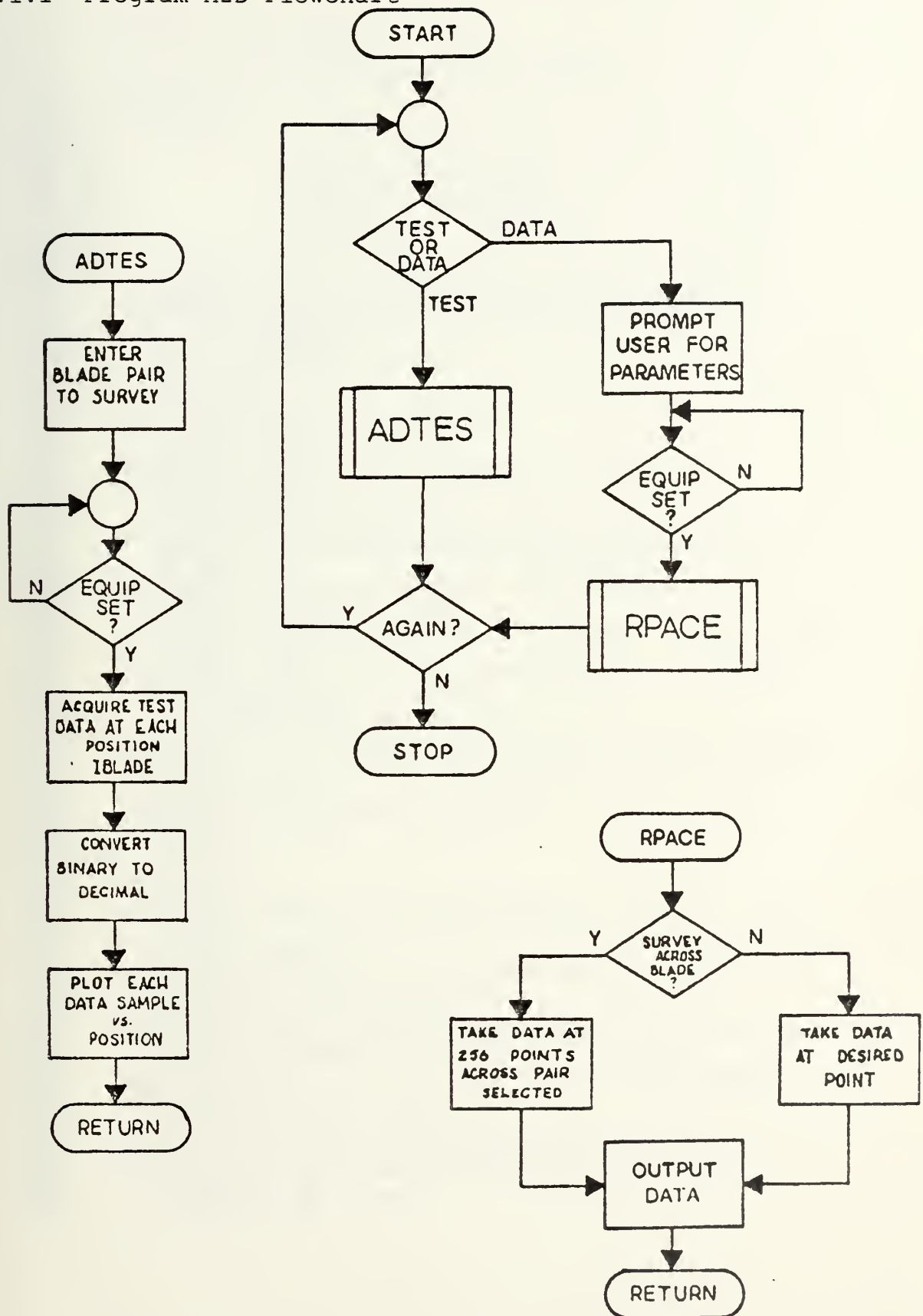
1. Copyright: The drivers DVR 70 and DVR 56 are copyrighted by the Hewlett-Packard Company, 1978. Approval for reproduction granted by Hewlett-Packard 22 May, 1981.
2. The driver flow chart in B.2.1 is a simplified diagram which shows the basic process for a typical driver. DVR 70 contains a series of steps which pass IBLADE (output) and a section which receives IRPM (inputs). The initiator section first outputs IBLADE to the PACER. After that, control is returned to the Central Interrupt Controller to await the PACER interrupt signal indicating it has IRPM ready to output. When the interrupt occurs, the completion section of DVR 70 is entered and IRPM is passed.

DVR 56, on the other hand, has only the input function to complete. It accomplishes this task as the standard driver indicated in the flow chart B.2.1. The beginning of DVR 56 configures the DMA feature of the RTE-IVB [Ref. 2].

B.1 ACQUISITION FORTRAN PROGRAM A2D

(See following pages).

B.1.1 Program A2D Flowchart



B.1.2 Program A2D Listing

```

AA2D      T=00004 IS ON CR00028 USING 00009 BLKS R=0000

0001      FTN4,L
0002      PROGRAM A2D
0003      C .....
0004      C .....
0005      C .....
0006      C .....
0007      C .....
0008      C .....
0009      C .....
0010      C .....
0011      C .....
0012      C .....
0013      COMMON IRPM
0014      INTEGER CHANL,AVERG,SURVEY,MODE,PAIR,POSIT,OFFSET
0015      90 WRITE (1,95)
0016      95 FORMAT(" WILL THIS BE SYSTEM TEST OR DATA RUN ?",
0017      * " 1=DATA 0=TEST")
0018      READ (1,*)ITEST
0019      IF (ITEST.EQ. 1) GO TO 96
0020      CALL ADTES(IGCB)
0021      CONTINUE
0022      GO TO 999
0023      96 WRITE (1,97)
0024      97 FORMAT(" ENTER TEST NUMBER ")
0025      READ (1,*)N2
0026      99 WRITE (1,100)
0027      100 FORMAT(" DO YOU WISH PROMPTING ? 1=YES 0=NO")
0028      READ (1,*)N1
0029      IF (N1.EQ. 1) GO TO 102
0030      101 WRITE (1,110)
0031      110 FORMAT(" ENTER CHANL,SURVEY,PAIR,POSIT,AVERG,OFFSET")
0032      READ (1,*) CHANL,SURVEY,PAIR,POSIT,AVERG,OFFSET
0033      GO TO 107
0034      102 WRITE (1,120)
0035      120 FORMAT(" ENTER DATA CHANNEL. LIMITS 0-15")
0036      READ (1,*)CHANL
0037      WRITE (1,112)
0038      112 FORMAT(" ENTER # DATA SAMPLES TO AVERAGE. LIMITS 1-99")
0039      READ (1,*) AVERG
0040      WRITE (1,130)
0041      130 FORMAT(" DO YOU WISH A SURVEY OR SINGLE PT ? 1=SURV",
0042      * " 0=SINGLE")
0043      READ (1,*) SURVEY
0044      IF (SURVEY.EQ. 1) GO TO 104
0045      WRITE (1,103)
0046      103 FORMAT(" WHICH BLADE PAIR DO YOU WISH TO SEE ? LIMITS",
0047      * " 1-9")
0048      READ (1,*) PAIR
0049      WRITE (1,113)
0050      113 FORMAT(" WHICH POSITION BETWEEN BLADE PAIR ? LIMITS ",
0051      * " 1-256")
0052      READ (1,*) POSIT
0053      MODE=1
0054      GO TO 107
0055      104 MODE=1
0056      WRITE (1,145)
0057      145 FORMAT(" WHICH BLADE PAIR DO YOU WISH TO SEE ? LIMITS",
0058      * " 1-9")
0059      READ (1,*) PAIR
0060      WRITE (1,165)
0061      165 FORMAT(" DO YOU WANT TO OFFSET THE SURVEY ? 1=YES 0=NO")
0062      READ (1,*) OFFSET
0063      IF (OFFSET.EQ. 0) GO TO 107
0064      WRITE (1,166)
0065      166 FORMAT(" ENTER % OFFSET. (WILL DELAY START % OF 256)"
0066      * " CHOICES- 50, 25, 12, OR 6")
0067      READ (1,*) OFFSET
0068      107 WRITE (1,170)
0069      170 FORMAT(" IS A/D CONVERTER ON ? IS TEST SET-UP READY ?",
0070      * " 1=YES 0=NO")
0071      READ (1,*)N8
0072      IF (N8.EQ. 0) GO TO 107
0073      175 CALL RPACE (CHANL,AVERG,SURVEY,MODE,PAIR,POSIT,OFFSET,N2)
0074      176 WRITE (1,177)
0075      177 FORMAT(" DO YOU WISH ANOTHER RUN ? 1=YES 0=NO")
0076      READ (1,*) N3
0077      IF (N3.EQ. 1) GO TO 90
0078      178 WRITE (6,168)

```



```

0079 160 FORMAT(25X,">>> END OF RUNS (<<"/))
0080 999 STOP
0081 END
0082 SUBROUTINE RSPACE (ICHAN,IAVG,ISURV,IMODE,IPAIR,
0083 CIPOSIT,IOTFS,N2)
0084 .....
0085 C .....
0086 C .....
0087 C .....
0088 C .....
0089 C .....
0090 C .....
0091 C .....
0092 C .....
0093 C .....
0094 C .....
0095 C .....
0096 C .....
0097 C .....
0098 C .....
0099 C .....
0100 C .....
0101 C .....
0102 C .....
0103 C .....
0104 C .....
0105 C .....
0106 C .....
0107 C .....
0108 C .....
0109 C .....
0110 C .....
0111 C .....
0112 C .....
0113 C .....
0114 C .....
0115 C .....
0116 C .....
0117 C .....
0118 C .....
0119 C .....
0120 C .....
0121 C .....
0122 C .....
0123 C .....
0124 C .....
0125 C .....
0126 C .....
0127 C .....
0128 C .....
0129 C .....
0130 C .....
0131 C .....
0132 C .....
0133 C .....
0134 C .....
0135 C .....
0136 C .....
0137 C .....
0138 C .....
0139 C .....
0140 C .....
0141 C .....
0142 C .....
0143 C .....
0144 C .....
0145 C .....
0146 C .....
0147 C .....
0148 C .....
0149 C .....
0150 C .....
0151 C .....
0152 C .....
0153 C .....
0154 C .....
0155 C .....
0156 C .....
0157 C .....
0158 C .....

160 FORMAT(25X,">>> END OF RUNS (<<"/))
999 STOP
END
SUBROUTINE RSPACE (ICHAN,IAVG,ISURV,IMODE,IPAIR,
CIPOSIT,IOTFS,N2)
.....
DATA ACQUISITION SUBROUTINE
.....
REAL SRVPT(256)
DIMENSION Ibuff(99),ITIME(5)
N=IAVG
CALL EXEC (11,ITIME,IYEAR)
WRITE (6,90) N2,ITIME(5),IYEAR
90 FORMAT(//,10X," THIS IS TEST #",I2," RUN ON JULIAN",
Y" DATE ",13,3X,I4,/)
IF (ISURV .EQ. 1) GO TO 120
.....
SINGLE POINT ACQUISITION
.....
IRLAD=256*(IPAIR-1)+IPOSIT
IF (IMODE .EQ. 0) GO TO 100
IRLAD=IRLAD+1000000
100 CALL EXEC (3,19)
CALL EXEC (1,19,IRPM,1,IRLAD)
CALL EXEC (1,20,IBUFF,N,ICHAN,0)
DO 110 I=1,IAVG
110 RIBUFF=RIBUFF+FLOAT(IBUFF(I))/32768.
PIDATA=RIBUFF/IAVG
GO TO 195
.....
SURVEY ACROSS BLADE PAIR ACQUISITION
.....
120 IF (IOFFS .EQ. 0) GO TO 125
IOFFS=100/IOFFS
GO TO 122
125 IOFFS=1
127 DO 140 J=1,256
IRLAD=256*(IPAIR-1)+J+256/IOFFS+1000000
CALL EXEC (3,19)
CALL EXEC (1,19,IRPM,1,IRLAD)
CALL EXEC (1,20,IBUFF,N,ICHAN,0)
RIBUFF=0.0
DO 130 K=1,N
130 RIBUFF=RIBUFF+FLOAT(IBUFF(I))/32768.
DATA=RIBUFF/IAVG
140 SRVPT(J)=DATA
.....
OUTPUT TABLES/PRINT
.....
WRITE (6,146)
146 FORMAT(//,23X,"PACED SURVEY DATA",/)
WRJ(E (6,148) SRVPT
148 FORMAT(8(2X,F8.7))
150 RPM=60/(IRPM*.000004)
WRITE (6,145) RPM
165 FORMAT(/,20X,"COMPRESSOR RPM FOR THIS RUN WAS ",
GF7.2,/)
GO TO 999
195 WRITE (6,196) IPOSIT,IPAIR,PIDATA
196 FORMAT(" THE DATA VALUE FOR POSITION ",I3," OF BLADE PAIR",
*12," IS ",F10.7)
GO TO 150
999 RETURN
END
SUBROUTINE ADTES(IGCB)
.....
TEST OF THE PACED DATA ACQUISITION
SYSTEM
.....
COMMON IRPM
DIMENSION IGCB(192)

```



```

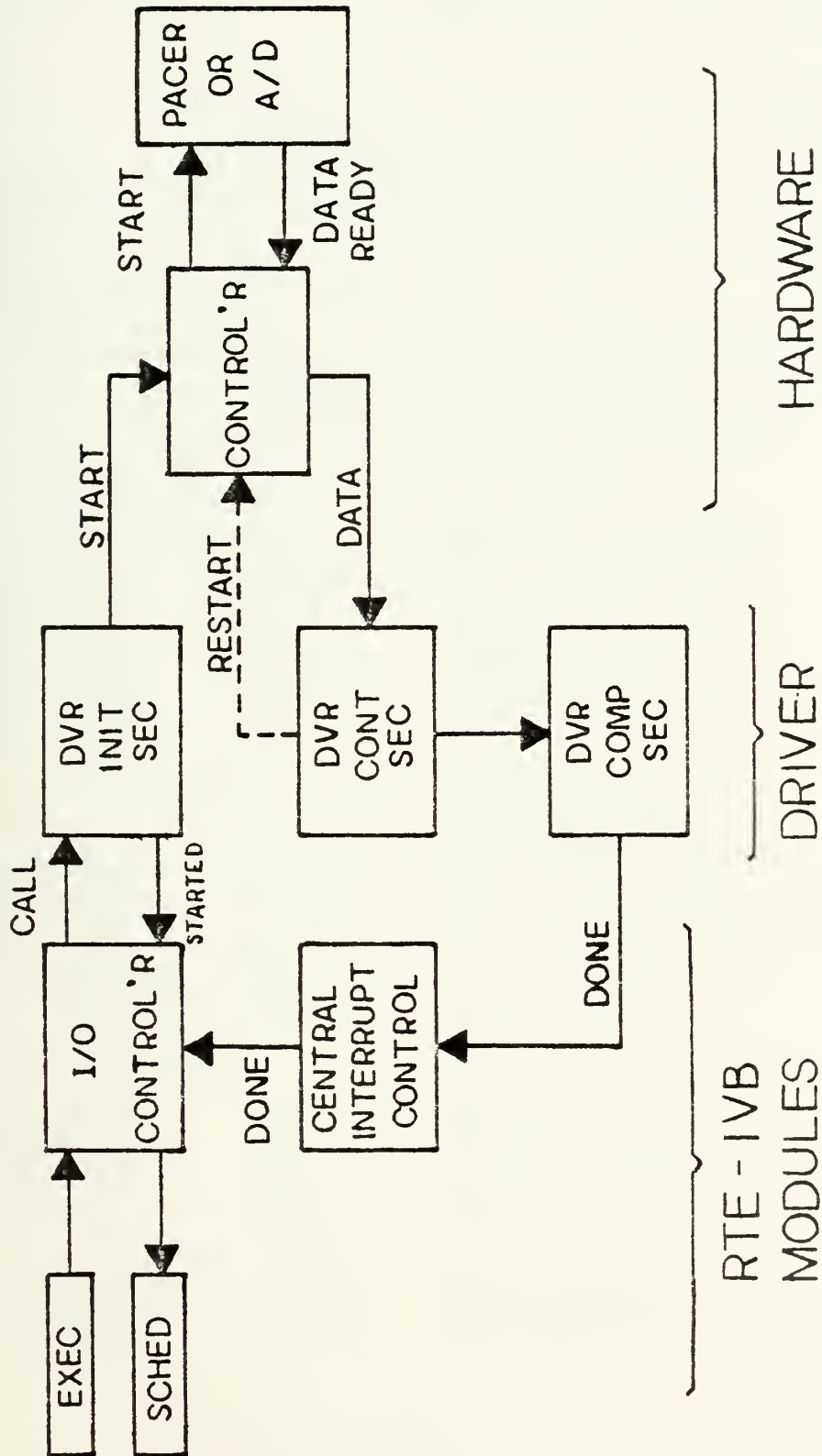
0159     INTEGER TIME(5),NOCR(2),IVOLT(128)
0160     REAL RVOLT(128)
0161     DATA NOCR /000033B,040433B/
0162     DATA ICHAN /0/
0163 101  FORMAT(" THIS WILL TEST THE PACED DATA SYSTEM FOR "
0164     *"CONTINUITY AND LINEARITY." /," ENTER THE SIMULATED",
0165     *" BLADE PAIR TO VIEW. LIMITS 1-8")
0166     WRITE (1,101)
0167     READ (1,*) IPAIR
0168 100  WRITE (1,102)
0169 102  FORMAT(" IS THE TEST SET UP READY AS PER MANUAL ?",
0170     *" 1=YES 0=NO")
0171     READ (1,*) N4
0172     IF (N4.EQ. 0) GO TO 100
0173     IBLADE = 0
0174     IBLADE = IBLADE+256*IPAIR
0175     IBLADE = IBLADE+1000008
0176     DO 128 I=1,128
0177     IBLADE=IBLADE+1
0178     CALL EXEC (3,19)
0179 20  CALL EXEC (1,19,IRPM,1,IBLADE)
0180 128 CALL EXEC(1,20,IVOLT(I),1,ICHAN,0)
0181     DO 45 J=1,128
0182     45 RVOLT(J)=FLOAT(IVOLT(J))/32768.
0183     LU=13
0184     ID=2
0185     CALL PLOTB(IGCB,ID,1,LU)
0186     CALL LIMIT(IGCB,0.,280.,0.,187.)
0187     CALL SETAR(IGCB,1.5)
0188     CALL VIEWP(IGCB,20.,140.,20.,80.)
0189     CALL WINDOW(IGCB,0.,128.,0.,1.)
0190     CALL FXD(IGCB,1)
0191     CALL LGRID(IGCB,-2.,.05,0.,0.,8.,5.,1.)
0192     CALL MOVE(IGCB,1.,RVOLT(1))
0193     DO 55 K=2,128
0194     EX=FLOAT(K)
0195 55  CALL DRAW(IGCB,EX,RVOLT(K))
0196     CALL VIEWP(IGCB,0.,150.,0.,100.)
0197     CALL WINDOW(IGCB,0.,150.,0.,100.)
0198     CALL MOVE(IGCB,64.,90.)
0199     CALL CPLOT(IGCB,-10,0.,0.)
0200     CALL LABEL(IGCB)
0201     WRITE(LU,160)
0202 160  FORMAT("PACED RAMP TEST DATA")
0203     CALL PLOTB(IGCB,ID,0)
0204     LU=6
0205     RETURN
0206     END

```


B.1.3 Program A2D Parameter Listing

CHANL/ICHAN	The A/D analog input channel to be sampled.
AVERG/IAVG	The number of samples per position to be averaged.
SURVEY/ISURV	Survey/single position selection
MODE/IMODE	Paced/free run-normally 1
PAIR/IPAIR	The pair of passages selected
POSI/IPOSIT	The position within the pair of passages
OFFSET/IOFFS	To start the survey later than position #1 within the pair passages. Entered as % of 256.
IRPM	See Table I
IBLADE	See Table I
IBUFF	The name of the set of digital data storage locations
N2	Test number that date
RBUFF	Floating point data storage
PTDATA/DATA	The data value at the selected point
SRVPT	The array holding the data surveyed
IGCB	Graphics control block, graphics package usage nonaccessible.

B.2 SOFTWARE DRIVERS



B.2.1 Flow Chart

B.2.2 Pacer Driver DVR 70

PAGE 0002 #01

8:35 AM FRI.. 4 AUG., 1978

```

0001          1548,R,L
0002 00000      NAM DVP70,0 NPGS RPACE RTE DRIVER REV. 780724 JDM
0003          ENT 1.70,C.70,C.XX,I.XX
0004*
0005* DRIVER FOR NAVAL POSTGRADUATE SCHOOL MOTEREY CA.
0006* AUTHOR: JIM MORRIS NEFLY SANTA CLARA 408-996-9800
0007*
0008* THIS RTE DRIVER WILL OUTPUT A BLADE NUMBER TO
0009* THE PACER AND RETURN THE RPM VALUE.
0010*
0011* CALLING SEQUENCE:
0012* CALL EXEC(1,LU,IRPM,LEN,IBLADE) >>>> NORMAL INPUT (READ)
0013*      LU      = LOGICAL UNIT NUMBER OF PACER
0014*      IRPM    = RETURNED RPM VALUE FROM PACER
0015*      LEN    = 1 (NORMAL), = 0 (NOP - IMMED. COMPLETION)
0016*      IBLADE = PACER BLADE NUMBER (16 BIT INTEGER)
0017*
0018* CALL EXEC(3,LU1) >>>> CLEAR CONTROL ON PACER
0019*
0020 00000 000000 1.70 NOP      ENTER INITIATION SECTION
0021 00001 0160640 JSR SETIO  CONFIGURE I/O INSTRUCTIONS
0022 00002 161665 LDA EQT6,I  GET CONTROL WORD
0023 00003 0121050 AND R3   ISOLATE REQUEST TYPE
0024 00004 0521060 CPA R1    INPUT?
0025 00005 0260030 IMP D.X1   YES - DOIT
0026 00006 0521050 CPA R3   CONTROL?
0027 00007 0260120 JMP CNTRL YES - DOIT
0028 00010 0621060 LDA R1    A=1 I.F. WRITE TO DEVICE NOT ILLEGAL
0029 00011 1260000 JMP 1.70,I  ERROR RETURN TO IOC
0030*
0031* CHECK FOR CLEAR CONTROL AND NO SUBFUNCTION BITS SET
0032*
0033 00012 161665 CNTRL LDA EQT6,I  GET CONTROL WORD
0034 00013 0121100 AND R3700 ISOLATE SUBFUNCTION BITS
0035 00014 0020002 SZA      ANY SET?
0036 00015 0260210 JMP REJCT YES, REJECT AS ILLEGAL CONTROL REQUEST
0037 00016 1067000 1.0 CLC SC   NO, CLEAR DEVICE AND RETURN
0038 00017 0621110 RT0N LDA R4    IMMEDIATE COMPLETION A=4
0039 00020 0020001 RSS      SKIP LOAD OF ERROR CODE A=2 (BAD CONTRL
0040 00021 0621070 REJCT LDA R2   REJECT ERROR A=2
0041 00022 1260000 JMP 1.70,I  RETURN TO IOC
0042*
0043* PROCESS READ REQUEST
0044*
0045 00023 161667 D.X1 LDA EQT8,I  GET BUFFER LENGTH
0046 00024 0020002 SZA      CHECK IF = 0
0047 00025 0260300 JMP D.X3   NO, NORMAL PROCESS (1 WORD WILL BE INPL
0048 00026 0064000 CLR      YES, B=0 (TRANSMISSION LOG)
0049 00027 0260170 JMP PT0N  RETURN TO IOC
0050*
0051* SETUP CONTINUATOR TO RETURN THIS SECTION
0052*
0053 00030 0621040 D.X3 LDA P2     ADJUST ADDRESS
0054 00031 0720350 STA C.70  STUFF INTO CONTINUATOR RETURN
0055 00032 0260460 JMP D.X2   ENTER CONTINUATOR SECTION

```



```

0056*
0057* NORMAL RETURN TO IOC NOW
0058*
0059 00033 002400 TEXIT CLA      A=0 (ALL IS WELL)
0060 00034 126000R JMP I.70,I    RETURN TO IOC
0061*
0062* CONTINUATION/COMPLETION SECTION
0063*
0064 00035 000000 C.70 NOP      ENTER CONT.
0065 00036 015064R JSR SETIO  CONFIGURE I/O
0066 00037 161660 LDA EQT1,I    CHECK FOR SPURIOUS INTERRUPT
0067 00040 112112R AND MASK    ISOLATE I/O REQUEST LIST POINTER
0068 00041 002002 SZA      IS A REQUEST IN PROGRESS?
0069 00042 026054R JMP I.3      YES, GO DO IT
0070 00043 171774 STA EQT15,I   NO, ZERO TIME-OUT CLOCK
0071 00044 036035R ISZ C.70    ADJUST RETURN TO P+2 (CONTINUATION)
0072 00045 126035R JMP C.70,I    RETURN TO CIC
0073*
0074* OUTPUT CONTROL WORD (BLADE NUMBER)
0075*
0076 00046 161670 D.Y2 LDA EQT9,I  GET BLADE NUMBER
0077 00047 000000 T.YX NOP      <<<<< DEBUG ENTRY POINT >>>>>
0078 00050 102600 I.1 OTA SC      OUTPUT TO DEVICE
0079 00051 103700 I.2 STC SC,C    TURN ON DEVICE
0080 00052 036035R ISZ C.70    ADJUST RETURN TO P+2 (CONTINUATION)
0081 00053 126035R JMP C.70,I    RETURN TO CIC
0082*
0083* COMPLETION SECTION
0084*
0085 00054 102570 I.3 LIA SC      GET RPM FROM PACER
0086 00055 165666 LDR EQT7,I  GET RPM BUFFER ADDRESS
0087 00056 000000 C.YX NOP      <<<<< DEBUG ENTRY POINT >>>>>
0088 00057 170001 STA B,I      STUFF INTO USER BUFFER
0089 00060 102400 CLA      SET A=0, ALL IS WELL RETURN CODE
0090 00061 006404 CLR,INB    SET B=1, TRANSMISSION LOG (1 WORD INPUT)
0091 00062 106700 I.4 CLC SC      CLEAR DEVICE
0092 00063 126035R JMP C.70,I    RETURN TO CIC, COMPLETE
0093*
0094* CONFIGURE I/O INSTRUCTIONS
0095*
0096 00064 000000 SETIO NOP      ENTRY TO SUBROUTINE
0097 00065 052102R CPA PIO     A=SC OF I/O DEVICE, CHECK IF CONFIGURED
0098 00066 126064R JMP SETIO,I YES, BYPASS CONFIGURATION
0099 00067 072102R STA PIO     SAVE CURRENT I/O CHANNEL NUMBER
0100 00070 032103R TOR LIA     COMBINE LIA WITH I/O
0101 00071 072054R STA I.3    STORE IT
0102 00072 042113R ADA B100   MAKE OTA INSTRUCTION
0103 00073 072050R STA I.1    STORE IT
0104 00074 042114R ADA B1100  MAKE STC,C INSTRUCTION
0105 00075 072051R STA I.2    STORE IT
0106 00076 032115R TOR B4000  MAKE CLC INSTRUCTION
0107 00077 072016R STA I.0    STORE IT
0108 00100 072062R STA I.4    AND AGAIN
0109 00101 126064R JMP SETIO,I RETURN FROM SUBROUTINE
0110*

```


111 CONSTANTS/STORAGE/LINKS

112

0113	00000	A	EQU	0	
0114	00001	R	EQU	1	
0115	00102	000000	P10	OCT	0
0116	00000	SC	EQU	0	CURRENT I/O SELECT CODE VALUE
0117	00103	102500	LIA	LIA	SC
0118	00104	0000320	P2	DEF	IEXIT-1
0119	00105	000003	R3	OCT	3
0120	00106	000001	R1	OCT	1
0121	00107	000002	R2	OCT	2
0122	00110	003700	R3700	OCT	3700
0123	00111	000004	R4	OCT	4
0124	00112	077777	MASK	OCT	77777
0125	00113	000100	R100	OCT	100
0126	00114	001100	R1100	OCT	1100
0127	00115	004000	R4000	OCT	4000

MASK OFF BIT 15

128

129 BASE PAGE COMMUNICATIONS AREA DEFINITIONS

130

0131	01650	:	EQU	16500	DEFINE START OF COMM AREA
0132	01660	FQT1	EQU	0+8	
0133	01661	FQT2	EQU	0+9	
0134	01662	FQT3	EQU	0+10	
0135	01663	FQT4	EQU	0+11	
0136	01664	FQT5	EQU	0+12	
0137	01665	FQT6	EQU	0+13	
0138	01666	FQT7	EQU	0+14	
0139	01667	FQT8	EQU	0+15	
0140	01670	FQT9	EQU	0+16	
0141	01671	FQT10	EQU	0+17	
0142	01672	FQT11	EQU	0+18	
0143	01771	FQT12	EQU	0+81	
0144	01772	FQT13	EQU	0+82	
0145	01773	FQT14	EQU	0+83	
0146	01774	FQT15	EQU	0+84	
0147	00116	SIZE	EQU	*	
0148		END			

** NO ERRORS *TOTAL **RTE ASMB 02067-16011**

B.2.3 A/D Driver DVR 56

ADVR56 T=00003 IS ON CR00002 USING 00024 BLKS R=0000

```

0001
0002 ASMB,R,L,B,C      DVR56      JUNE,71
0003 HED {2310/2311 SUBSYSTEMS RTE DRIVER}
0004 NAM DVR56
0005 ENT I.56,C.56
0006 SPC 1
0007 * FORTRAN CALL: CALL EXEC (1,IDRT,IBUFF,N,ICHAN,ICODE)
0008 * IDRT SUBSYSTEM DEVICE REFERENCE NUMBER
0009 * IBUFF INTEGER ARRAY (DATA STORAGE BUFFER)
0010 * N NUMBER OF CONVERSIONS (DATA POINTS)
0011 * ICHAN CHANNEL NUMBER
0012 * ICCDE: SUBSYSTEM/MODE:
0013 * 0 2311 DIG ENCODE
0014 * 1 2311 DIG PACE
0015 * 2 2311 SEQ ENCODE
0016 * 3 2311 SEQ PACE
0017 * 4 2311 DIG FREE
0018 * 5 2311 SEQ FREE
0019 * 6 2310 DIG
0020 * 7 2310 SEQ
0021 SPC 1
0022 * INITIATION SECTION
0023 SPC 1
0024 I.56 NOP
0025 SPC 1
0026 * CONFIGURE INITIATION SECTION IO
0027 SPC 1
0028 STA B SAVE IO ADDRESS
0029 IOR OTA CONFIGURE
0030 STA IO12
0031 STA IO1 A/D
0032 STA IO13
0033 STA IO9 CONVERTER
0034 STA IO15
0035 STA IO16
0036 ADA =B300 IO
0037 STA IO7 INSTRUCTIONS
0038 ADA =B600
0039 STA IO10
0040 STA IO14
0041 LDA CHAN
0042 IOR OTA
0043 STA IO2 CONFIGURE
0044 ADA =B1100 DMA
0045 STA IO8 IO
0046 ADA =B176774 INSTRUCTIONS
0047 STA IO5
0048 ADA =B4000
0049 STA IO3
0050 XOR =B4100
0051 STA IO4
0052 STA IO6
0053 ADA =B4104
0054 STA IO11
0055 SKP
0056 * VALID REQUEST CHECK
0057 SPC 1
0058 LDA EQT6,I READ
0059 CPA =B1
0060 JMP *+3 YES
0061 ERROR CLA,INA NO - REJECT
0062 JMP I.56,I RETURN
0063 LDA EQT8,I NUMBER OF REQUESTED
0064 CMA,INA DATA POINTS GREATER
0065 SSA,RSS THAN ZERO?
0066 JMP ERROR NO - GO TO REJECT
0067 SPC 1
0068 * CONSTRUCT DMA CONTROL WORD
0069 SPC 1
0070 CLA,CCE INITIALIZE SWITCH
0071 STA W1011 TO 2310 OPERATION
0072 LDA B IO ADDRESS INTO A
0073 ADA =B20000 ADD CLC OPTION
0074 LDB EQT10,I CODE WORD INTO B
0075 ADE =D-6 6 OR 7? I.E., 2310?
0076 CCE,SSB YES
0077 JMP ,2311 NO, 2311 OPERATION
0078 STB D0.S1 SET TO 2310 SEQ OR DIG MODE

```



```

0079      ELA,RAR          ADD STC OPTION
0080      JMP I02
0081      SPC 1
0082      .2311 STB W1011   SET SWITCH TO 2311 OPERATION
0083      LDB EQT10,I      CODE WORD INTO B
0084      CPB =B2          IF CODE 0 OR 2,
0085                      ADD STC OPTION
0086      SZB,RSS
0087      ELA,RAR
0088      STA TEMP          SAVE DMA CONTROL WORD
0089      SPC 1
0090      * CONSTRUCT A/D CONTROL WORD
0091      SPC 1
0092      LDA EQT9,I        CHANNEL # TO A
0093      I016  OTA A.2.D
0094      CPB =B4          ICODE  COMMAND  PROGRAM
0095      JMP AHEAD          *      *      *
0096      RBR,SLB          0      DIG ENCODE  0000CH
0097      JMP,#+4          1      DIG PACE    0100CH
0098      RBR,SLB          2      SEQ ENCODE  040000
0099      JMP,#+3          3      SEQ PACE    050000
0100      JMP SINGL        4      DIG FREE   0200CH
0101      RBR              5      SEQ FREE   070000
0102      SPC 1
0103      CLA
0104      I01  OTA A.2.D    RESET A/D CONVERTER
0105      LDA =B40000
0106      SPC 1
0107      SINGL RBL
0108      SSB
0109      ADA =B10000      PACER ENABLE BIT
0110      RBR,SLB
0111      AHEAD ADA =B20000  FREE RUN BIT
0112      STA B
0113      SKP
0114      * OUTPUT COMMAND WORDS TO DMA AND A/D
0115      SPC 1
0116      LDA TEMP
0117      I02  OTA DMA      CW1 TO DMA
0118      I03  CLC DMA
0119      LDA EQT7,I        BUFFER ADDRESS TO A
0120      IOR =B100000      DMA INPUT BIT
0121      I04  OTA DMA      CW2 TO DMA
0122      I05  STC DMA
0123      LDA EQT8,I        WORD COUNT (BUFF LENGTH)
0124      CMA,INA          NEGATIVE TO OUTPUT TO DMA
0125      I06  OTA DMA      CW3 TO DMA
0126      CLF 0            TURN OFF INTERRUPT
0127      LDA W1011        2310 OR 2311 OPERATION?
0128      CCE,SZA,RSS      2311
0129      JMP,2310          2310
0130      LDA B
0131      I07  CLF A.2.D
0132      I09  OTA A.2.D    ACTIVATE
0133      I010 STC A.2.D     A/D CONVERTER
0134      I08  STC DMA,C    ACTIVATE DMA
0135      CLA
0136      CPA DUMMY
0137      JMP I.56,I        RETURN
0138      I011 CLC DMA
0139      LDB INTEA
0140      LDA CHAN
0141      CPA =D7
0142      INB
0143      LDA B,I
0144      IOR =B100000
0145      STA B,I
0146      STF 0
0147      CLA              NORMAL
0148      JMP I.56,I        RETURN
0149      SPC 1
0150      .2310 LDA EQT9,I   CHANNEL # TO A
0151      I012  OTA A.2.D     OUTPUT RANDOM MODE
0152      ELA,RAR          SET DIGITIZE MODE
0153      I013  OTA A.2.D     OUTPUT DIG MODE
0154      I014  STC A.2.D,C   ACTIVATE 2310
0155      LDB D0.S1        DIG OR SEQ?
0156      SZB,RSS
0157      JMP I015          DIGITIZE
0158      CLE,INA          SEQUENTIAL

```



```

0159      ELA,RAR
0160      IOR =B40000      SET SEQ MODE
0161      IO15  OTA A.2.D      OUTPUT NEXT MODE
0162      JMP  IO8
0163      SKP
0164      * COMPLETION SECTION
0165      SPC 1
0166      C.56  NOP
0167      IOR CLC      CONFIGURE
0168      STA *+1      CLC DMA INSTRUCTION
0169      CLC DMA
0170      LDA EQT4,I      A/D ADDRESS
0171      AND =B77      TO A
0172      IOR OTA      CONFIGURE OTA A.2.D
0173      STA *+2      INSTRUCTION
0174      CLA,CCE      TURN OFF
0175      OTA A.2.D      PACER
0176      LDB EQT8,I      TRANSMISSION LOG
0177      ELB,RBR      TO B
0178      JMP C.56,I      RETURN COMPLETION
0179      SKP
0180      * CONSTANTS
0181      CLC CLC IO
0182      OTA OTA IO
0183      D0.S1 BSS 1
0184      W1011 BSS 1
0185      SPC 3
0186      IO EQU 0
0187      A.2.D EQU 0
0188      DMA EQU 0
0189      TEMP EQU C.56
0190      A EQU 0
0191      B EQU 1
0192      SPC 3
0193      * SYSTEM BASE PAGE COMMUNICATION AREA
0194      SPC 1
0195      . EQU 1650B
0196      SPC 1
0197      EQT4 EQU .+11
0198      EQT6 EQU .+13
0199      EQT7 EQU .+14
0200      EQT8 EQU .+15
0201      EQT9 EQU .+16
0202      EQT10 EQU .+17
0203      SPC 1
0204      CHAN EQU .+19
0205      INTBA EQU .+4
0206      DUMMY EQU .+55
0207      SPC 3
0208      END

```


APPENDIX C
PACED DATA ACQUISITION
USERS MANUAL

The two sections of this Appendix describe the use of program A2D for both (C.1) System Verification and (C.2) Test Data Acquisition.

C.1 SYSTEM VERIFICATION

In order to verify the complete paced data acquisition system (software and hardware), the following steps should be followed using the equipment shown in Fig. C.1.

C.1.1 Procedure

A WaveTek 142 signal generator or equivalent should be used to drive the test pulse feature of the PACER.

- (1) Connect the "sync" output of the signal generator to the "sync" input on the PACER panel (Fig. 7).
- (2) Connect the 50 Ω output of the WaveTek to the A/D analog channel to be tested (normally 0) and to the oscilloscope.
- (3) Turn on the A/D converter.
- (4) Set the WaveTek panel switches to produce a ramp voltage of 1 volt maximum peak amplitude from the 50 Ω output.
- (5) On the PACER front panel connect the jack marked "BL" INPUT to the jack marked "BL" OUTPUT. Do

the same for the jacks marked "REV" INPUT and "REV" OUTPUT.

- (6) Make sure "PACER ON" switches are in the "ON" position.
- (7) Ensure that the Card #3 with the frequency range encompassing the blade passing frequency set on the WaveTek generator is installed in the PACER. If necessary remove the front panel air vent and replace Card #3 with the proper range card. Card #3 is shown in Figure 7.
- (8) Turn on the PACER power switch and verify that the red pilot lamp is lit on the front panel.
- (9) Log on the 21MX computer following the directions in the TPL Data Acquisition Manual.
- (10) Once logged on, mount cartridge 28. Turn on the plotter and select the desired pen. Call up the Acquisition (Fortran) Program A2D with the command RP, A2D. Run the program with the command RU, A2D. The interactive program will prompt the user for responses. The responses are explained in the prompts which are given at the terminal. The prompts are as follows:
 - (a) System test or data run: enter Ø.
 - (b) Simulated blade pair to survey: enter any number 1-8.
 - (c) Is test set-up ready: if yes-enter 1, if no-enter Ø.

After prompt (c) is answered yes, and if the test is successful, the plotter will plot the same ramp signal that was set on the oscilloscope in C.1.1 step 4 (Fig. 12). The linearity and smoothness of the ramp signal indicate the degree to which data acquired under pacer control agree with the analog data input to the A/D converter.

C.2 TEST DATA ACQUISITION

In order to acquire paced data from the compressor (or other) test rig, the following steps should be followed with the equipment shown in Fig. C.1.

C.2.1 Procedure

- (1) Cables to the PACER from the optical timing wheel on the test machine should be connected as shown in Fig. C.2. Verify the transducer input connections to the A/D converter at the A/D junction box.
- (2) Turn on the A/D converter. Turn on the signal conditioner.
- (3) Log on the 21MX computer following directions in the TPL Data Acquisition User's Manual. Call the Acquisition (Fortran) Program A2D by using the command RP, A2D. Then run the program by issuing the command RU, A2D.
- (4) The interactive program will prompt the user for the following:

- (a) System Test or Data Run: enter 1.
- (b) Test number - enter integer.
- (c) Do you wish prompting: Yes - enter 1,
no - enter Ø.

From this point on, the program prompts are self-explanatory.

- (5) At the completion of the data acquisition, the data values are printed out as shown in Table C-I.
- (6) The final prompt will ask if another run is desired.

C.2.2 Data Storage

The survey data acquired in the program A2D is contained in the data memory locations SRUPT (J) where $J = 1 - 256$. The program A2D may be modified to output the data as desired by the user or to pass the data to a user-written subroutine for analysis.

Table C-1. Paced Data Output from Program A2D
THIS IS TEST # 5 RUN ON JULIAN DATE 149 1981

PACED SURVEY DATA

0742188	0937500	1093750	1386719	1582031	1738281	1894531
1253125	2148438	2500000	3636719	3734375	2929688	3046875
3203125	3359375	3652344	3808594	3925781	4140625	4277344
4114063	4570313	4746094	5019531	5097656	5312500	5488281
5605469	5742188	5917969	6074219	6386719	6523438	6660156
6835938	6972656	7050781	7207031	7402344	7734375	7832031
7988281	8164063	8320313	8515625	8828125	8964844	9042969
9179688	9335938	9492188	9667969	9980469	9824219	9746094
9570313	9737500	9921875	9765625	8769531	8613281	8535156
8320313	8144531	8046875	7753906	7558594	7402344	7324219
7109375	6972656	6796875	6523438	6347656	6250000	6074219
5937500	5742188	5585938	5468750	5175781	5078125	4921875
4667969	4609375	4453125	4238281	3945313	3808594	3691406
3535156	3417969	3222656	3066406	2753906	2636719	2460938
2363281	2187500	1992188	1875000	1582031	1445313	1269531
1074219	1015625	0859375	0703125	0410156	0488281	0664063
0761719	0937500	1132813	1210938	1542969	1679688	1855469
1992188	2128906	2285156	2480469	2753906	2890625	3027344
3203125	3359375	3457031	3593750	3945313	4160156	4238281
4375000	4531250	4726563	4804688	5097656	5351563	5410156
5585938	5781250	5820313	6093750	6367188	6523438	6582031
6738281	6933594	7109375	7226563	7597656	7675781	7871094
7968750	8144531	8281250	8457031	8750000	8925781	9042969
9218750	9570313	9570313	9687500	9960938	9902344	9765625
9355469	9218750	9218750	9082031	8710938	8513281	8437500
8203125	8046875	8046875	7871094	7617188	7460938	7265625
6972656	6835938	6835938	6699219	6406250	6232813	6132813
5761719	5644531	5429688	5273438	5195313	5097656	4960938
4550781	4414063	4257813	4179688	3945313	3847656	3652344
3398438	3222656	3085938	2968750	2792969	2625781	2441406
2187500	2031250	1855469	1699219	1542969	1425781	1269531
0996094	0859375	0859375	0566406	0429688	0449219	0644531

COMPRESSOR RPM FOR THIS RUN WAS 18541.4

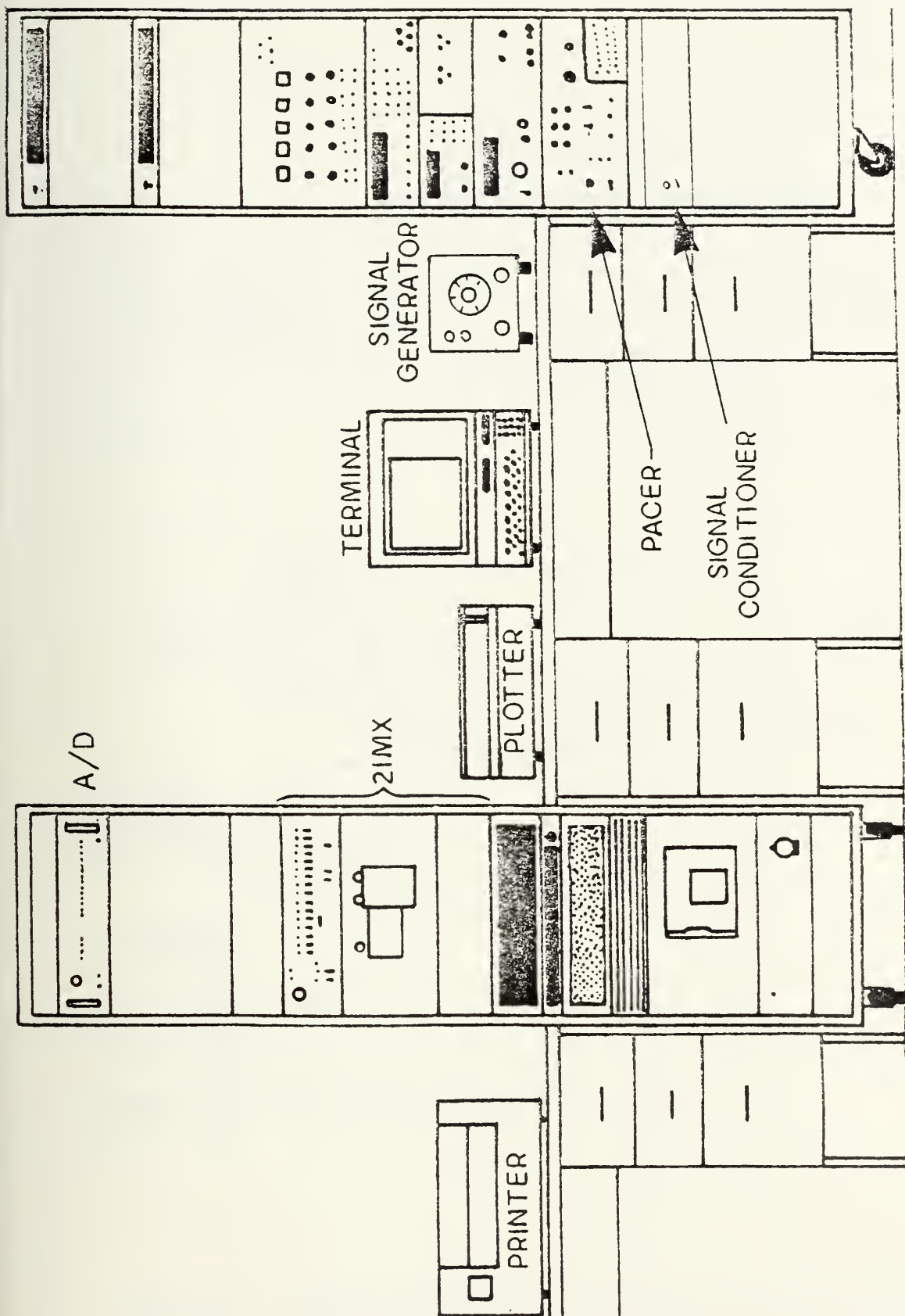


Figure C1. Data Acquisition and Test Equipment

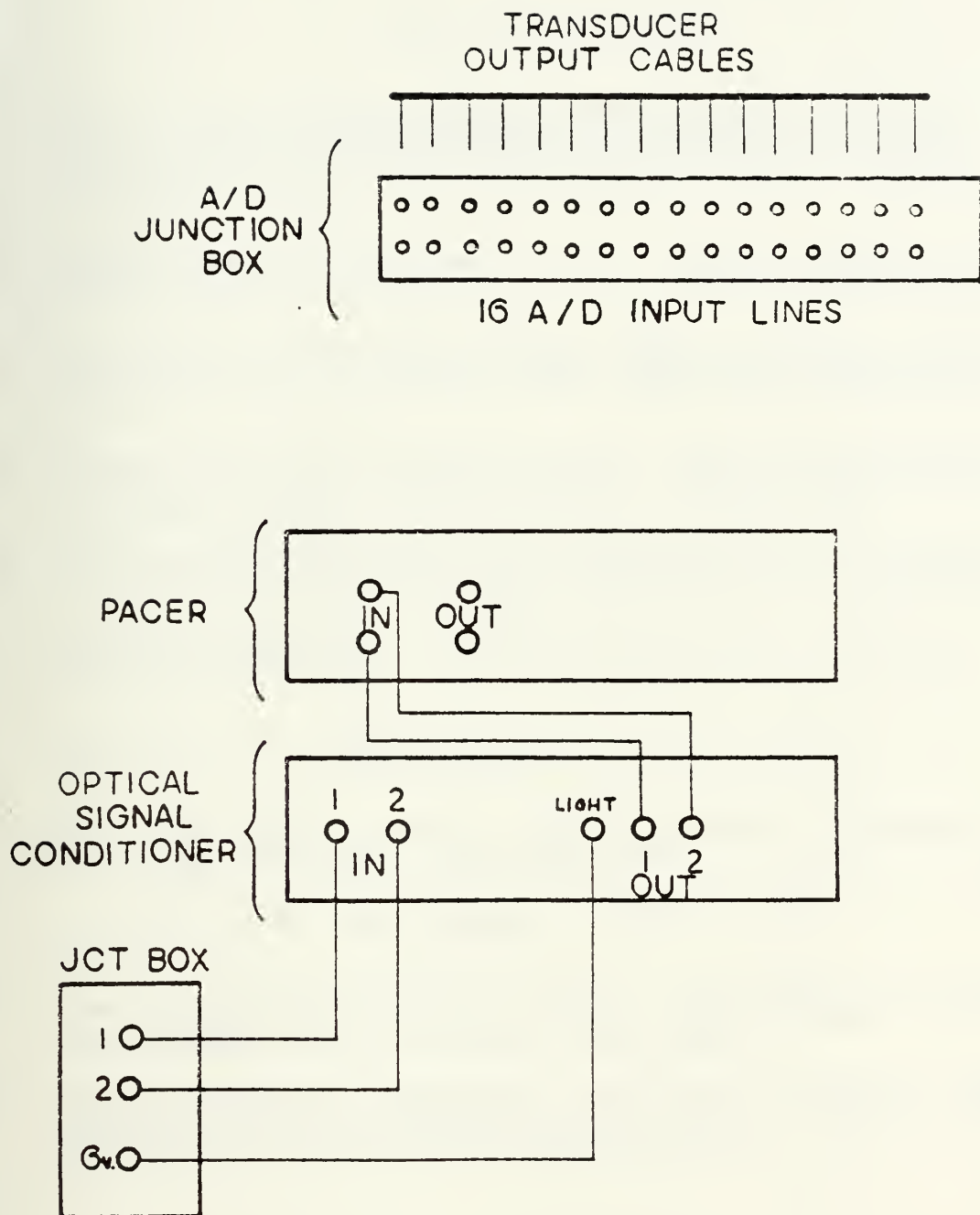


Figure C2. Cable Connections for Test Data Acquisition

LIST OF REFERENCES

1. West, J. C., Digital Programmable Timing Device for Fast Response Instrumentation in Rotating Machines, M.S.A.E. Thesis, Naval Postgraduate School, Monterey, California, 1976.
2. Hewlett-Packard Reference Manual, HP21MX Computer Series, HP02108-90002, June 1976.
3. Geopfarth, R. N. Introductory Guide for Users of RTE-IV, TPL Technical Note 79-01, TPL, Monterey, California, 1979.
4. Hewlett-Packard Reference Manual, RTE-IVB Programmer's Reference Manual, HP92068-90004, January 1980.
5. Hewlett-Packard Operating and Service Manual, Analog to Digital Converter, 5610A, 05610-91999, January 1972.
6. Signetics Catalog, Signetics Corporation, Copyright 1977.
7. Lancaster, D., CMOS Cookbook, Howard W. Sams & Co. Inc., 1977.
8. RCA Solid State Databook, COS/MOS Digital Integrated Circuits, SSD-203B, Copyright 1973.
9. Lancaster, D., TTL Cookbook, Howard W. Sams & Co. Inc., 1974.
10. Hewlett-Packard Reference Manual, RTE FORTRAN IV Reference Manual, HP92060-90023, March 1980.
11. Hewlett-Packard Programming and Operating Manual, RTE Operating System Drivers Writing Manual, HP92200-93005, May 1978.

INITIAL DISTRIBUTION LIST

	No. Copies
1. Defense Technical Information Center Cameron Station Alexandria, Virginia 22314	2
2. Library, Code 0142 Naval Postgraduate School Monterey, CA 93940	2
3. Department Chairman, Code 67 Department of Aeronautics Naval Postgraduate School Monterey, CA 93940	1
4. Director, Turbopropulsion Laboratory, Code 67Sf Naval Postgraduate School Monterey, CA 93940	1
5. LCDR J. C. West 6229 Hanley Corpus Christi, TX 78412	1
6. Turbopropulsion Laboratory Code 67 Naval Postgraduate School Monterey, CA 93940	15
7. Dr. Gerhard Heiche Naval Air Systems Command Code AIR-310 Department of the Navy Washington, D.C. 20360	1
8. Dr. A. D. Wood Office of Naval Research Eastern/Central Regional Office 666 Summer Street Boston, MA 02210	1
9. LCDR P. A. McCarville 8151 Alton Dr. Lemon Grove, CA 92045	1

Thesis

M1666

McCarville

c.1

Hardware and software improvements to a paced data acquisition system for turbo-machines.

193686

Thesis

M1666

McCarville

c.1

Hardware and software improvements to a paced data acquisition system for turbo-machines.

193686

thesM1666

Hardware and software improvements to a



3 2768 002 12313 5

DUDLEY KNOX LIBRARY